



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2020

Geological map generalization driven by size constraints

Sayidov, Azimjon ; Aliakbarian, Meysam ; Weibel, Robert

Abstract: Geological maps are an important information source used in the support of activities relating to mining, earth resources, hazards, and environmental studies. Owing to the complexity of this particular map type, the process of geological map generalization has not been comprehensively addressed, and thus a complete automated system for geological map generalization is not yet available. In particular, while in other areas of map generalization constraint-based techniques have become the prevailing approach in the past two decades, generalization methods for geological maps have rarely adopted this approach. This paper seeks to fill this gap by presenting a methodology for the automation of geological map generalization that builds on size constraints (i.e., constraints that deal with the minimum area and distance relations in individual or pairs of map features). The methodology starts by modeling relevant size constraints and then uses a workflow consisting of generalization operators that respond to violations of size constraints (elimination/selection, enlargement, aggregation, and displacement) as well as algorithms to implement these operators. We show that the automation of geological map generalization is possible using constraint-based modeling, leading to improved process control compared to current approaches. However, we also show the limitations of an approach that is solely based on size constraints and identify extensions for a more complete workflow.

DOI: <https://doi.org/10.3390/ijgi9040284>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-187303>

Journal Article

Published Version



The following work is licensed under a Creative Commons: Attribution 4.0 International (CC BY 4.0) License.

Originally published at:

Sayidov, Azimjon; Aliakbarian, Meysam; Weibel, Robert (2020). Geological map generalization driven by size constraints. *ISPRS International Journal of Geo-Information*, 9(4):284.

DOI: <https://doi.org/10.3390/ijgi9040284>

Article

Geological Map Generalization Driven by Size Constraints

Azimjon Sayidov , Meysam Aliakbarian and Robert Weibel 

Department of Geography, University of Zurich, Winterthurerstrasse 190, 8057 Zurich, Switzerland; meysam.aliakbarian@geo.uzh.ch (M.A.); robert.weibel@geo.uzh.ch (R.W.)

* Correspondence: azimjon.sayidov@geo.uzh.ch; Tel.: +41-44-635-51-98

Received: 14 February 2020; Accepted: 22 April 2020; Published: 24 April 2020



Abstract: Geological maps are an important information source used in the support of activities relating to mining, earth resources, hazards, and environmental studies. Owing to the complexity of this particular map type, the process of geological map generalization has not been comprehensively addressed, and thus a complete automated system for geological map generalization is not yet available. In particular, while in other areas of map generalization constraint-based techniques have become the prevailing approach in the past two decades, generalization methods for geological maps have rarely adopted this approach. This paper seeks to fill this gap by presenting a methodology for the automation of geological map generalization that builds on size constraints (i.e., constraints that deal with the minimum area and distance relations in individual or pairs of map features). The methodology starts by modeling relevant size constraints and then uses a workflow consisting of generalization operators that respond to violations of size constraints (elimination/selection, enlargement, aggregation, and displacement) as well as algorithms to implement these operators. We show that the automation of geological map generalization is possible using constraint-based modeling, leading to improved process control compared to current approaches. However, we also show the limitations of an approach that is solely based on size constraints and identify extensions for a more complete workflow.

Keywords: geological maps; map generalization; constraint-based methods; size constraints; generalization operators

1. Introduction

The majority of the research conducted on the automation of map generalization has thus far been dedicated to topographic maps, with a focus on the generalization of roads, boundaries, river networks, and buildings; that is, with a focus on linear objects or small area objects. The generalization of thematic map types, including categorical maps (e.g., geological, soil, or land use maps), has received less attention, perhaps since categorical maps contain polygons of potentially arbitrary shapes and sizes, rendering them more complex than typical shapes found, for instance, for buildings on a topographic map. Nevertheless, categorical maps, with geological maps as a prominent representative, are a frequent map type and require specific methods with which to automate generalization. For instance, in topographic maps, buildings are usually of a regular shape and are often arranged in a regular fashion (e.g., in linear alignments), while in categorical maps, polygon features can be of an arbitrary shape, occurring in arbitrary spatial arrangements. Merely reusing the approach and processes of topographic map generalization for categorical mapping will not provide a proper solution [1,2], as requirements for categorical and thus geological map generalization are different from those used for topographic mapping, as detailed in Section 2.

Over the past two decades, constraint-based techniques have evolved as the leading paradigm used to model and automate the map generalization process [3–6] and to evaluate generalization results [7]. However, while use of the constraint-based paradigm is wide-spread in topographic mapping, generalization methods for categorical and geological maps have rarely adopted this approach (see [8] for one exception). This is somewhat disappointing, as the complexity of categorical maps could favor an approach to allow differentiated and adaptive modeling and monitoring of the conditions that govern the map generalization process.

The objective of this paper is to present a methodology using constraints, more specifically so-called size constraints, as a basis for the geological map generalization. Size constraints deal with the minimum area and distance relations in individual or pairs of map features. They have also been termed metric or graphical constraints [4] and express the natural limits of human perception in measurable, minimal dimensions [8]. The proposed approach first identifies a list of size constraints, their goal values, and measures; it also prioritizes the logical treatment of constraints, which in turn dictates the sequence of generalization operators and algorithms to be used in case a constraint is violated. The main driver for the proposed methodology is the “minimum area” (MA) constraint, which influences other constraints and is coupled with descriptive attributes of the polygon features that are being generalized. In an adaptive workflow, the MA and related constraints are successively tested and, if violated, trigger appropriate generalization operators, including object elimination (i.e., selection), enlargement, aggregation, and displacement.

The use of the constraint-based approach in map generalization has predominantly been linked to the agent-based paradigm [3,5,6,8,9]. Since building an agent engine, however, is by no means a trivial task, the uptake of the agent-based approach, and with it the constraint-based approach, in practice has been limited. Hence, the main contribution of this research is the demonstration of the usage of constraints, with a focus on size constraints, for the automated generalization of geological maps, as this approach has so far not been explicitly applied to the automated generalization of geological maps. Furthermore, we use a workflow-based approach consisting of a sequence of several generalization operators, as this has a better potential to be adopted in practice. Although designed with geological maps in mind, the approach is also applicable to other categorical maps: maps that are entirely covered with polygonal features (i.e., so-called polygonal subdivisions) such as soil maps, vegetation maps, or land-use and land-cover maps. In experiments, we show that the proposed methodology, despite its relative simplicity, is capable of more appropriately generalizing geological maps, with better local control over the generalization operations that are applied as compared to results generated by state-of-the-art solutions that do not use a constraint-based approach. Finally, we are aware that with its focus on size constraints only, our methodology has limitations, which we point out in Section 6, providing leads for further research.

In Section 2, the characteristics of geological maps are discussed. Section 3 provides a review of related work on geological map generalization. Section 4 represents the core of the article, introducing the elements of the size constraint-based methodology one by one. Section 5 presents the results of a series of experiments, firstly to illustrate the effect of the individual generalization operators, and then for different parameterizations of the combined workflow. Finally, Section 6 provides a discussion of the experimental results, while Section 7 ends the paper with conclusions and an outlook.

2. Geological Maps: Purpose and Peculiarities

Geological maps are among the most complex thematic maps, with various elaborate shapes and structures. This renders the generalization process more demanding, and thus an in-depth analysis of these structures is required before the generalization process [2,10]. Creating geological maps requires not only cartographic expertise but also general geological knowledge. Thus, for instance, knowledge on the formation and structure of rock types occurring in a study area will be crucial when it comes to identifying the relative significance of map features. However, the importance of certain map features may vary depending on the purpose and type of geological map that is to be made.

Bedrocks assist geologists in portraying the natural history of a study area and identifying associated rock formations. They also carry essential mineral resources such as coal, oil, and uranium, which are in the focus of the mining industry. Thus, a map for mining purposes highlights ancient bedrocks that may carry particular minerals, while neglecting sedimentary rocks as they are more recent [11]. Geophysicists, in turn, place more emphasis on the intrinsic characteristics of features such as porosity and permeability in rock and sediments [12].

The goal of a geological map is “to interpretively portray the spatial and temporal interactions between rocks, earth materials, and landforms at the Earth’s surface” [13]. Geological materials are the igneous, metamorphic, and sedimentary rock and surficial sediments that form the landscape around us. Most geological maps use colors and labels to show areas of different geological materials, called geological units. Geological structures are the breaks and beds in the geological structures resulting from the slow but strong forces that form the world [14]. “Geological maps show the location of these structures with different types of lines. Because the Earth is complex, no two maps show the same materials and structures, and so the meaning of the colors, labels, and lines is explained on each map” [15].

Geological maps consist of diverse patterns formed by a fabric of polygons, plus additional linear objects (e.g., fault lines) and point objects (e.g., wells) which however are not of concern here. The polygons can be described by spatial, structural, and semantic properties to evaluate similarities or differences and thus infer spatial patterns relating to the perceptual structure or arrangement of polygon features on the map [16]. Figure 1 shows some sample excerpts of geological maps, ordered by increasing geometric and graphical complexity. In the simple map extract of Figure 1a, few geological units are involved, with relatively simple shapes, which could be generalized by merely using simplification operators. The next level of complexity comprises many small polygons of the same or similar geological units (Figure 1b), which may be generalized by removing or alternatively aggregating units or sub-units into a single group. Another level consists of a series of elongated polygons of the same unit embedded in, and possibly crossing, other units (Figure 1c), where a cartographer may recommend merging neighboring units while trying to maintain their overall arrangement (i.e., using the so-called typification operator). Another complicated form found in geological maps are tree-like, dendritic forms, which were created at a later stage of the quaternary period by rivers and streams carrying sediments and other minerals. This type also defines the position of a river system (Figure 1d). In this case, a possible solution is to replace several branches with a smaller number of simplified tree branches. Figure 1e shows that various kinds of units consisting of small and large/long and narrow polygons and tree-like structures may also co-exist, rendering the generalization process even more challenging. The generalization of such complex fabrics requires making multiple, interrelated (and possibly conflicting) generalization decisions.

The few examples in Figure 1 illustrate that the polygonal layer of geological maps consists of a far greater variability in category, size, shape, boundary sinuosity, and topological (in particular containment) relations of the concerned polygons than can typically be found on topographic maps. The spatial arrangement of polygons in geological maps can take many different forms, as is noticeable in Figure 1, although even in the complex map of Figure 1e we can perceive alignments and clusters of polygons. The key feature classes in topographic maps are predominantly anthropogenic and hence tend to have more regular shapes and a lesser degree of variability, and they are often arranged in regular alignments (e.g., grid street networks or straight alignments of buildings). “Natural” feature classes, such as land cover, in topographic maps usually are restricted to few categories (e.g., woodlands, waterbodies, built-up areas) and are of secondary priority. Hence, we would argue that while the same operators (elimination, simplification, aggregation, typification displacement, etc.) are valid in both domains, different generalization algorithms, or at least different combinations of algorithms, have to be used to adapt to the peculiarities of geological maps or more generally categorical maps.

The next section offers a review of various generalization methods that have been explicitly proposed to deal with the peculiarities of the geological maps listed above.

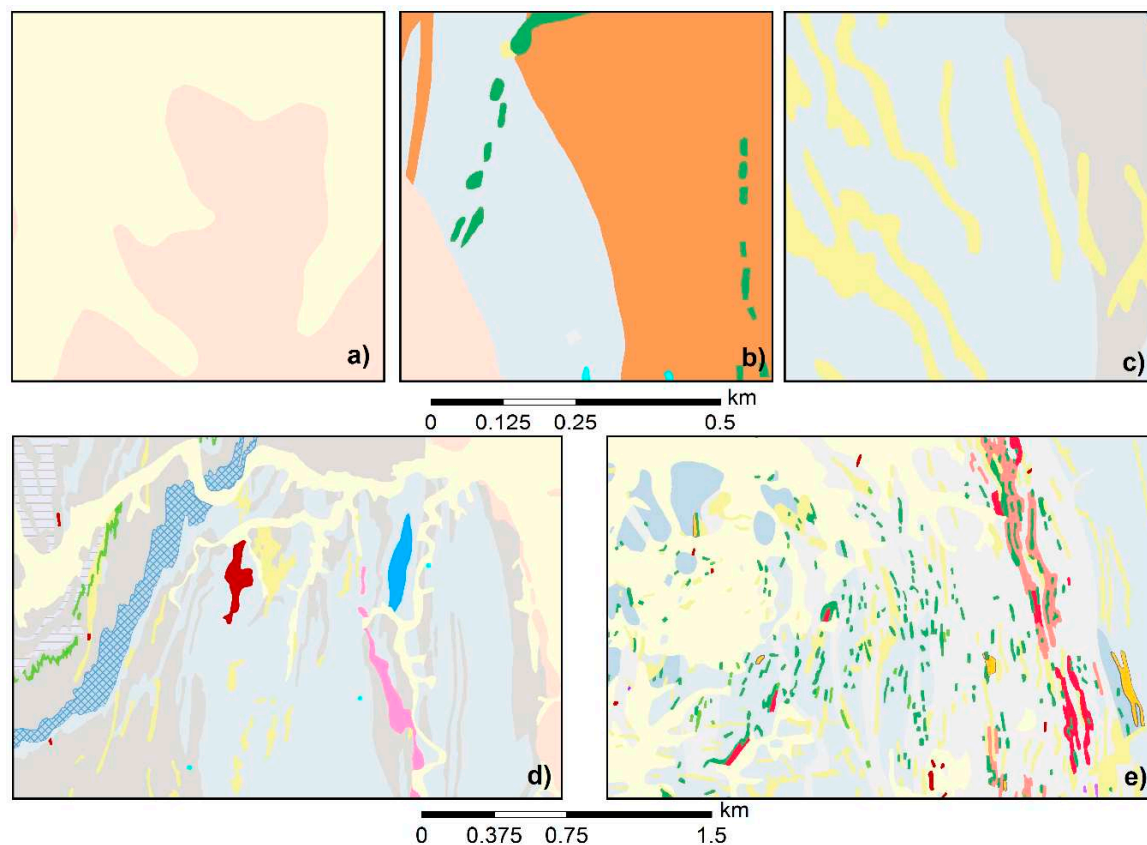


Figure 1. Some excerpts from geological maps, ordered by the complexity of the shapes and structures of geological units. (a) A few large polygons with simple shapes. (b) Several small polygons of the same geological unit and with similar geometrical properties. (c) Elongated polygons of the same unit with similar geometrical properties. (d) A complex type with tree-like dendritic shapes. (e) A combination of all of the above-mentioned shapes and structures.

3. Related Work

The earliest significant attempt at specifically generalizing geological maps was made by the authors of [17], who tried to automate the generalization of a 1:250,000 scale geological map from a 1:50,000 scale source map (a product of the British Geological Survey (BGS)), using the conceptual model previously suggested in [18]. While the results obtained were encouraging, the BGS concluded that the strategy still required intervention by the human operator, rendering it less flexible and more subjective. Importantly, however, this early work highlighted the significance of basing the generalization process upon an understanding of the essential structures and patterns inherent to the source map, a task that is of course not unique to the case of geological maps [18,19].

The authors of [20] presented a conceptual workflow model dedicated to the semi-automated generalization of thematic maps with three main phases: structural analysis, generalization, and visualization. Structural analysis (or “structure recognition” according to [18]) was deemed especially crucial, as once all relevant structures present in a map are made known, this information can support the decision of “when and how to generalize” [21]. The second step of their conceptual model consisted of constraint modeling in a multi-agent system, aiming to build an objective and flexible workflow.

Inspired by the aforementioned conceptual model [20], the author of [22] developed a generalization workflow based on ArcGIS tools. A sample geological map at a 1:24,000 scale was generalized to three target scales, namely 1:50,000, 1:100,000, and 1:250,000. The study results were compared with the corresponding U.S. Geological Survey (USGS) geological maps and helped

to summarize the strengths and limitations of the tools available for generalization at the time within ArcGIS.

Another experiment on the effectiveness of ArcGIS was conducted by Smirnoff et al. [23], where a cellular automata (CA) approach was developed specifically for geological map generalization using ArcGIS tools as a basis. When comparing their results with those of a process using the generalization tools directly available in ArcGIS, they concluded that the cell-based, or cellular, automata model had essential advantages for the automated generalization of geological maps. In more recent research [24], the ArcGIS toolbox called “GeoScaler” was tested on surficial and bedrock maps. The results were evaluated and found adequate. Also, repeatable results were obtained by maintaining some amount of human intervention in the process. Very importantly, since the GeoScaler toolbox was made freely available, this methodology can be used in the practice of geological mapping and has thus defined the state of the art of generalization tools for geological map generalization, which persists today. However, the approach does not consider the individual, local properties of geological features such as the size, shape, and orientation of the polygons, or the distance between them, which is crucial for carrying over the critical patterns of the source map to the derived map. Moreover, as most geological map data exist in vector format, conversion from vector to raster format (and possibly back to vector format again) causes additional, uncontrolled loss of data accuracy.

Hence, a vector-based approach that uses a combination of generalization operators that can be adaptively applied depending on the local situation seems more appropriate. In this vein, the authors of [25] proposed algorithms for several operators of polygon generalization, including elimination, enlargement, aggregation, and displacement, based on a rolling ball principle, conforming Delaunay triangulation, and skeleton approaches. The authors of [8,26] proposed a list of constraints and an agent-based process for the generalization of polygonal maps, extending earlier works by the authors of [4,5,9].

Probably the most comprehensive work regarding constraint modeling for geological maps was presented by Edwardes and Mackaness [27], who developed and illustrated a rich set of constraints and proposed ways to represent these in formal language. While conceptually intriguing, the proposed set of constraints was unfortunately not linked to particular generalization algorithms and was not implemented. The study also demonstrated the vast complexity that is involved when trying to model the constraints governing geological maps comprehensively and cast these into a computer-based process.

Müller and Wang [28] proposed an automated methodology for area patch generalization that encompassed elimination, enlargement/contraction, merging, and displacement operators. They addressed a problem similar to that dealt with in this paper, and they also used a similar set of operators. However, their approach considered all polygons as semantically identical. In contrast, geological maps easily consist of over 20 rock types, demanding simultaneous consideration of geometrical as well as attribute properties of the polygons. Their approach also leaves rather little flexibility for modifying the control parameters, as it lacks the capability of testing different solutions for a given conflict.

As argued in Section 1, the best way to detect cartographic conflicts and formalize and control generalization algorithms for resolving such situations is by using constraints, as has been shown in topographic map generalization. In this paper, we seek to demonstrate the application of the constraint-based paradigm to the case of geological maps. As shown in Section 2 and another study [27], there is an almost infinite complexity involved when trying to solve geological map generalization comprehensively. Thus, we focus on a particular generalization problem: small, free-standing polygons (area patches in the terminology of [28]), which represent a frequent case in geological maps (Figure 1b,d,e) and other types of categorical maps, such as soil maps. Since such small polygons are prone to legibility problems in scale reduction but also often represent geological units of superior economic value (Section 4.4.2), we focus on size constraints, which are simple yet allow many of the problems tied to small polygons to be addressed. A limited set of rather simple constraints alleviates tracing the effects of using a constraint-based approach in polygonal map generalization.

4. Methodology

4.1. Overall Workflow

An overview of the proposed methodology is shown in Figure 2. The overall workflow starts from a large-scale, detailed map stored in a geological database (at scales of 1:10,000 to 1:25,000) to obtain generalized maps at medium scales (1:50,000 to 1:100,000), subsequently storing them again in the database. The actual generalization workflow consists of three main stages: identification of constraints, modeling of constraints, and generalization execution. While this part of the workflow shares some similarity with the model presented in [29], the focus here is simply on representing the sequence of the main stages. In the initial stage (Section 4.2), the relevant cartographic constraints governing geological maps are identified, in our case size constraints related to the generalization of small area patches. This is followed by the stage of constraint modeling (Section 4.3), which aims to formalize the said constraints for geological map generalization. The third and main stage is the actual generalization process (Section 4.4), which takes the previously modeled constraints and uses them to orchestrate the application of different generalization operators—elimination, enlargement, aggregation, displacement—that can be used to resolve potential constraint violations. Lastly, the obtained map is evaluated based on the goal values of each representative constraint. If the map is regarded as not satisfactory by the set goal values, control is returned to the constraint modeling stage to explore alternative options, optimizing the importance and priority values of the constraints. Once the resulting map is considered satisfactory, it is stored in the database.

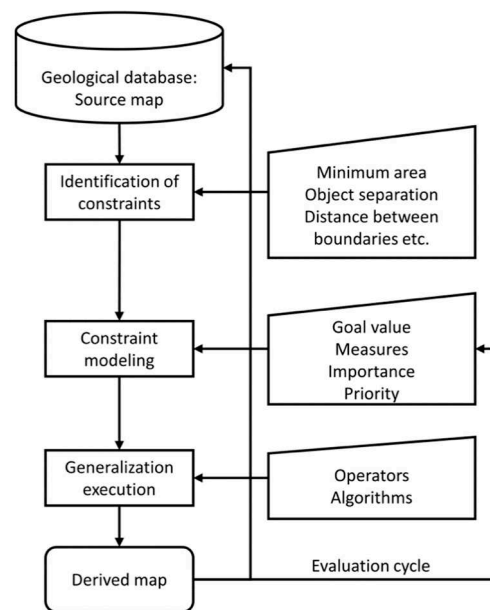


Figure 2. The overall workflow of the proposed methodology.

4.2. Identification of Constraints

As we focus on the generalization of small area patches in categorical maps, the relevant set of constraints is restricted to size constraints. Size constraints form the very starting point of the generalization process, as they deal with minimum area and distance relations in individual map features or pairs of map features. Therefore, while they may be considered to be more straightforward than, for instance, constraints dealing with groups of multiple polygons (expressing spatial pattern configurations), they are nonetheless of crucial importance as they form the foundation of all other constraints. The primary trigger of these constraints is the illegibility of map objects due to scale reduction when the limits of human visual perception are reached and polygons become too small

to be visible, or are too close, causing visual coalescence. Thus, two constraints form the core of size constraints: “minimum area” (MA), and “object separation” (OS). Based on the study of relevant literature [8,30,31], we identified the following five size constraints for our generalization of geological maps:

- Minimum area
- Object separation
- Distance between boundaries
- Consecutive vertices
- Outline granularity

4.3. Constraint Modeling

Once the constraints have been identified, they need to be further defined. Following the approach proposed in [8], which follows the agent-based paradigm, a complete constraint definition consists of the constraint itself, one or more goal values (minimum dimension to be attained on the target map), one or more measures that can quantify whether the constraint is met, possible plans (i.e., generalization operators triggered in case of constraint violation), and the importance or priority of generalization operators.

Table 1 summarizes the five size constraints, the cause that may lead to their violation, goal values, measures, possible generalization operators, and the impact that will be caused by generalization. Figure 3 illustrates the size constraints visually.

Table 1. List of size constraints and associated elements. Goal values are taken from [30] but could be modified (cf. Section 5.6).

Constraint	Cause	Goal Value	Measures	Plans/Possible Operators	Impact
Minimum area	Scale reduction	0.5 × 0.5 mm	Area	Elimination	Area loss and gain (area constraint), change of the overall configuration of polygons
				Enlargement	Change of minimum distance between features
				Aggregation	Loss of overall spatial pattern, shape distortion
Object separation	Scale reduction, enlargement	0.4 mm	Shortest distance	Displacement	Minimum distance between features, positional accuracy
				Enlargement, exaggeration	Shape distortion, ratio between features
				Aggregation	Loss of overall spatial pattern, shape distortion
				Typification	Loss of overall spatial pattern, shape distortion
Distance between boundaries	Scale reduction	0.6 mm	Internal buffer	Enlargement	Minimum distance between features
Consecutive vertices	Scale reduction	0.1 mm	Shortest distance between vertices	Elimination	Shape distortion polygon outline
Outline granularity—width	Scale reduction	0.6 mm	Shortest distance	Simplification, smoothing	Shape distortion
Outline granularity—height	Scale reduction	0.4 mm	Shortest distance	Simplification, smoothing	Shape distortion

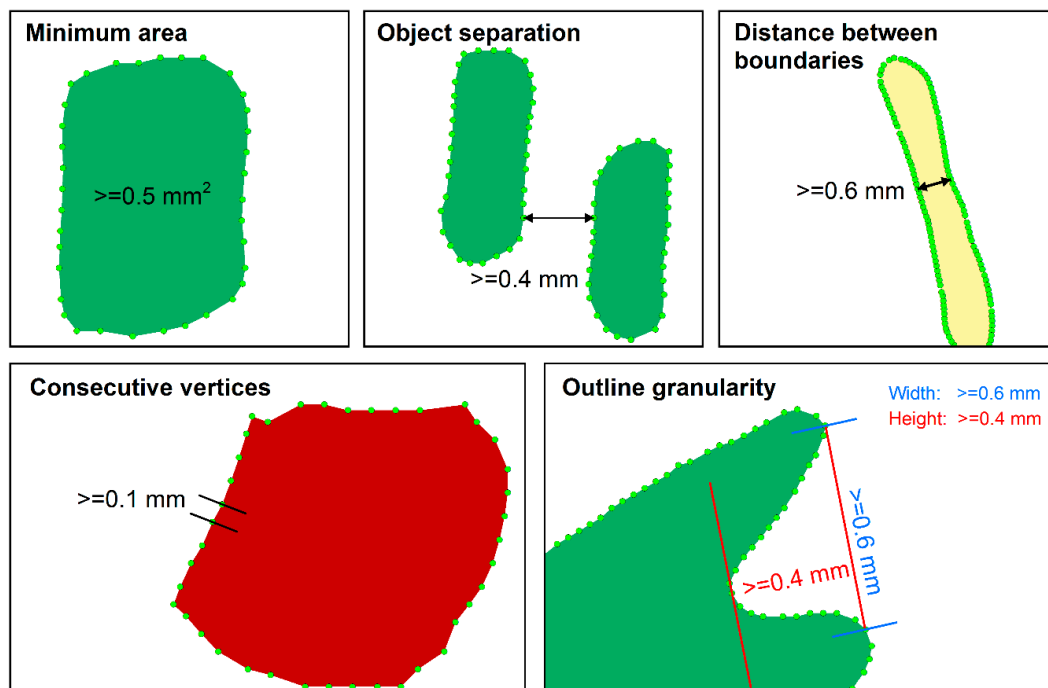


Figure 3. Graphical representation of the size constraints.

4.4. Generalization Execution

4.4.1. Generalization Workflow

The minimum area constraint is fundamental to the generalization of polygonal maps, as every type of categorical map is related to the size of polygonal features. Hence, it serves as the initial trigger and decision point for the generalization process, which is modeled after a typical process of map generalization in traditional cartography [32]. The workflow in Figure 4 is represented in two main branches, but it is essentially sequential. First, the right-hand branch deals with polygons that are too small. If they do not belong to an essential geological unit, they are eliminated. Else, they are enlarged, using a different enlargement algorithm depending on their shape. After having dealt with the elimination and enlargement operators, the workflow returns to check if now every polygon is large enough. If the answer is yes, the left-hand branch is triggered, which deals with polygons larger than the minimum area limit. Here, the next size constraint is tested. When the minimum object separation distance to its neighbors is not met, if the polygons concerned are of the same category they are aggregated or else displaced to meet the object separation constraint. Else, the polygon can be output to the target map or database. This final task also includes further line simplification and smoothing operations to take care of the “consecutive vertices” and “outline granularity” constraints.

In the remainder of this section, we will introduce the different generalization operators and algorithms one by one. Note that we assume that any reclassification operations merging sub-categories into superordinate categories take place before our workflow.

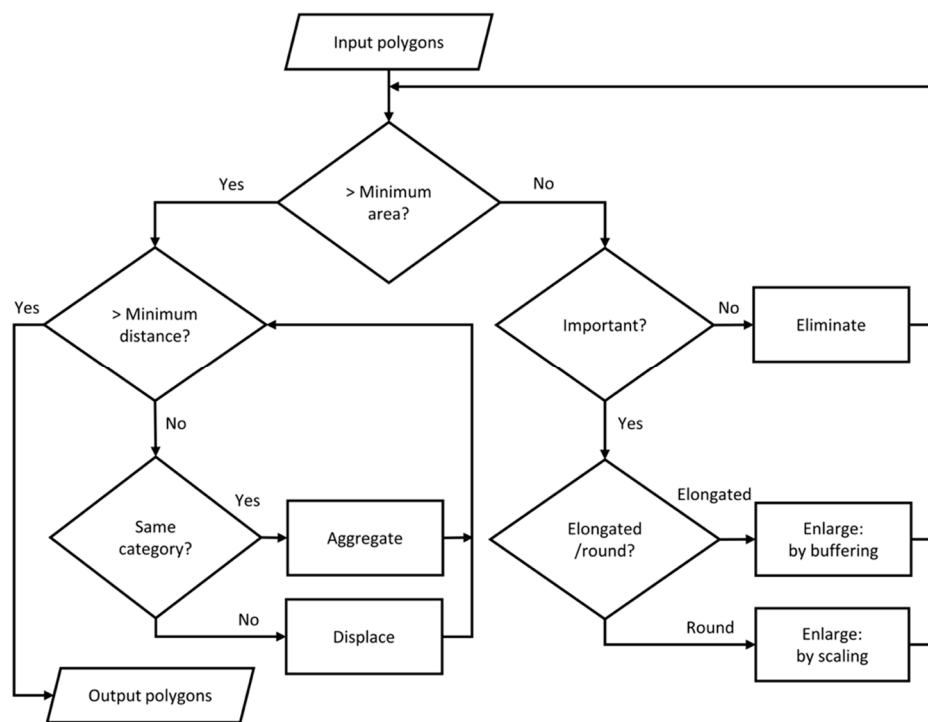


Figure 4. The workflow of the polygon generalization stage, governed by size constraints.

4.4.2. Elimination

The elimination operator dissolves small polygons into background polygons. It is triggered after a polygon fails two tests: it does not meet the MA constraint, and it is considered not sufficiently important to be nevertheless retained. However, the two tests are not applied in an unbounded way on the polygons. Still, a global target value is first determined, giving the number of polygons to be eliminated (n_{elim}) from the source map (or conversely selected for the target map). Three different selection methods are devised, all factoring in both the MA constraint and the importance of polygons given by their category (i.e., the geological unit in our case).

The first selection method, the “Radical Law” [33], is used as a baseline, as it is often used as a reference in map generalization. Consequently, the number of polygons to be eliminated (n_{elim}) is determined using the Radical Law. A detailed example of how n_{elim} is determined is given in Section 5.2. Once n_{elim} has been established, the polygons to be removed must be identified. The polygons falling below the MA threshold are sorted in ascending order according to the sum of their normalized area and their normalized geological importance value. The importance of a polygon is a function of its geological unit, which reflects geological age. Older units, such as “amphibolite” and “pegmatite”, will have more mineral deposits and hence more economic value than newly generated rocks [11]; younger rock types, such as “quaternary”, are thus considered less critical than ancient “Jurassic” layers. After the sorting order has been established, the n_{elim} smallest and least important polygons are then eliminated.

In the second selection method, called “area loss–gain selection”, the polygons are first sorted by category in ascending order of their area, and polygon removal then also proceeds by category (i.e., by geological unit). In each category, as many small polygons are removed as are needed to balance the area gain of the remaining same-class, small polygons treated by the enlargement operator (next subsection). Thus, the total area of the removed polygons per category is always balanced with the area gained by the enlargement operator. For that purpose, this selection operator for each polygon pre-computes the area gain that would be added if the polygon was kept and enlarged.

The third approach, termed as “category-wise selection”, applies the Radical Law separately for each category, as indicated by its name. For instance, if a map has 5 categories and according to the Radical Law 30% of polygons should be removed, then in each category 30% of the smallest polygons are eliminated. Note that this method does not take into account the area loss and gain of the polygons, yet it better balances the removal of polygons from each category than the pure Radical Law does, protecting categories consisting only of small polygons from vanishing.

4.4.3. Enlargement

After removing small and unimportant polygons in the elimination step, some polygons will remain that are smaller than the MA limit, but owing to their importance are flagged for retention. Therefore, the next step is to enlarge them until they reach the MA limit, and adequate readability by the human eye is ensured. Two algorithms are devised to accomplish enlargement, buffering, and scaling. Figure 5 illustrates the effect of these two algorithms: enlargement by buffering generates round polygon shapes while scaling enlarges the polygon area and maintains the initial polygon shape.

Polygon buffering is a commonly used operation in GIS and is straightforward to accomplish. However, its tendency to generate round shapes may lead to the loss of the polygon’s original shape, mainly if very small polygons are involved and/or the buffer distance is significant. Hence, we recommend using buffer enlargement when only a smaller increase is necessary, and when polygons have a more elongated shape, where shape distortion is less noticeable.

The scaling algorithm can handle round polygons and extend small polygons without distortion of the shape. The algorithm loops over every vertex and calculates its distance to the point of reference, which is the polygon centroid by default. The scaled polygon is then produced by extending a line from the reference point across each initial vertex at the scaled length, with the endpoint becoming the scaled vertex. One disadvantage of the basic scaling algorithm is that when the polygon centroid is used as a reference point, topological errors may be created for non-convex polygons (Figure 5c), necessitating displacement to restore the topology. The enlarged polygon in Figure 5c is shifted from its original position to the right-hand side and below due to the scaling process.

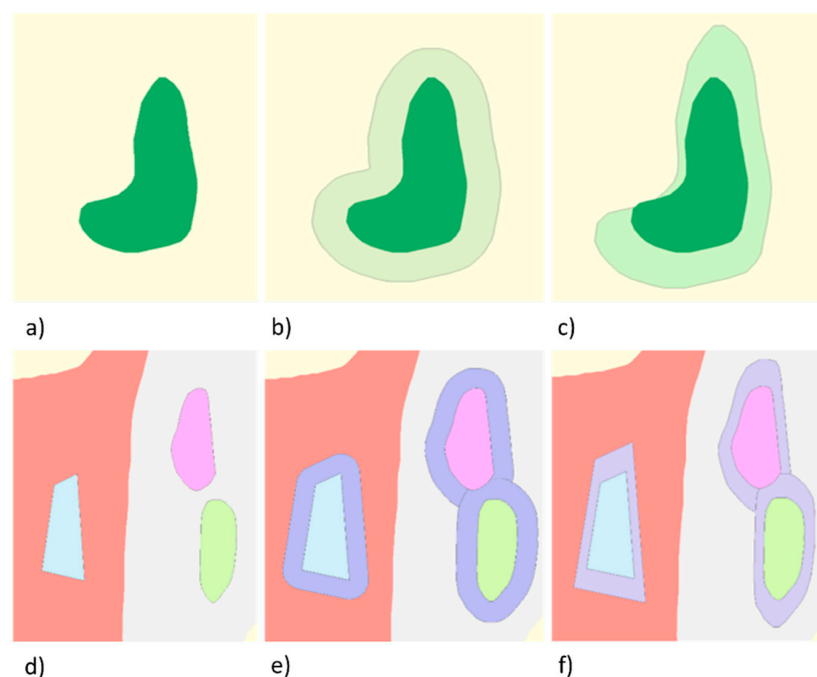


Figure 5. Enlargement. (a,d) The original polygons. (b,e) Those enlarged by the buffering algorithm. (c,f) Those enlarged by the scaling algorithm.

The choice between the two enlargement algorithms for each polygon is guided by an analysis of their shape using the “ipsometric quotient” (IPQ) shape index [34], used as a measure of compactness in [35]:

$$IPQ = 4\pi \times \frac{A}{P^2} \quad (1)$$

where A denotes the area and P the perimeter of the polygon. The IPQ takes values in the interval (0,1), where values approaching 0 indicate elongated shapes, while a value of 1 indicates a circle. We used a threshold of 0.5 in our experiments to distinguish between elongated shapes (for buffering) and round shapes (for scaling), as this threshold marks the midpoint between the two extremes.

The buffer width K_b that needs to be added to the polygon is calculated as a function of the target area A' to be reached at the target scale (e.g., 625 m² at 1:50,000), the current area A , and current perimeter P

$$K_b = \frac{A' - A}{P} \quad (2)$$

The scaling factor K_s that allows the algorithm to enlarge a polygon to the target area A' (e.g., 625 m² at 1:50,000) equals the square root from the target area A' over the current area A .

$$K_s = \sqrt{\frac{A'}{A}} \quad (3)$$

4.4.4. Aggregation

After execution of the enlargement operator, all remaining polygons should be sufficiently large for the target scale and thus adequately readable. However, enlargement also may give rise to another conflict: the violation of the object separation constraint, which is dealt with the aggregation and displacement operator, respectively, in this work. Both are triggered by a violation of object separation, but aggregation resolves this problem by merging polygons that are too close but also similar, while displacement moves dissimilar polygons until the minimum separability distance is again met. A conflict between features is identified by calculating the distance between any two polygons, where the polygons are closest to each other.

The similarity of polygons may be determined using several properties, such as the shape, size, orientation, and category of involved polygons [10], but in this paper, we use only the semantic similarity, that is, same category, to denote two polygons as being similar. The semantic similarity is of overriding importance: it makes only sense to aggregate polygons of same categories (or possibly of shared subcategories, if they exist).

Various algorithms have been proposed for the aggregation and typification of buildings [36,37], but for our purposes, we were more interested in an algorithm that would fill in voids between polygons that are found to be too close, akin to the algorithm in [25]. The aggregation is performed by generating a concave hull from the outline vertices of the polygons to be aggregated using the algorithm by [38]. The inset maps of Figure 12 highlight sample results of this procedure. The algorithm starts by selecting the first vertex as the one with the smallest Y value. Next, k points nearest to the current vertex are selected as candidates to be the next vertex for the output polygon. The next vertex is decided based on the biggest right-hand turn from the horizontal axis. These two steps are repeated until the selected candidate is the first vertex. Finally, the consecutive vertices found are connected by line segments to form the resulting concave polygon. Adjusting the parameter k allows for control of the degree of concavity of the resulting aggregated polygon.

4.4.5. Displacement

There are two reasons why the displacement operator may be activated. First, as a direct consequence of scale reduction, visually differentiating between map objects becomes harder and at some stage, impossible (an effect often termed congestion; [18]). Second, as a consequence of polygon

enlargement, the object separation constraint may be violated, as enlarged polygons require more space and may start to coalesce or even overlap. In both cases, map objects should be either aggregated or displaced to the minimum separability range. In instances where polygons are densely clustered, alternative operators such as typification (which is a combination of elimination, aggregation, and displacement) may be preferred.

Figure 6 illustrates the effect of the displacement operator, triggered by a violation of the object separation constraint following polygon enlargement. After the violation has been detected, polygons are moved to the minimum separability distance from each other.

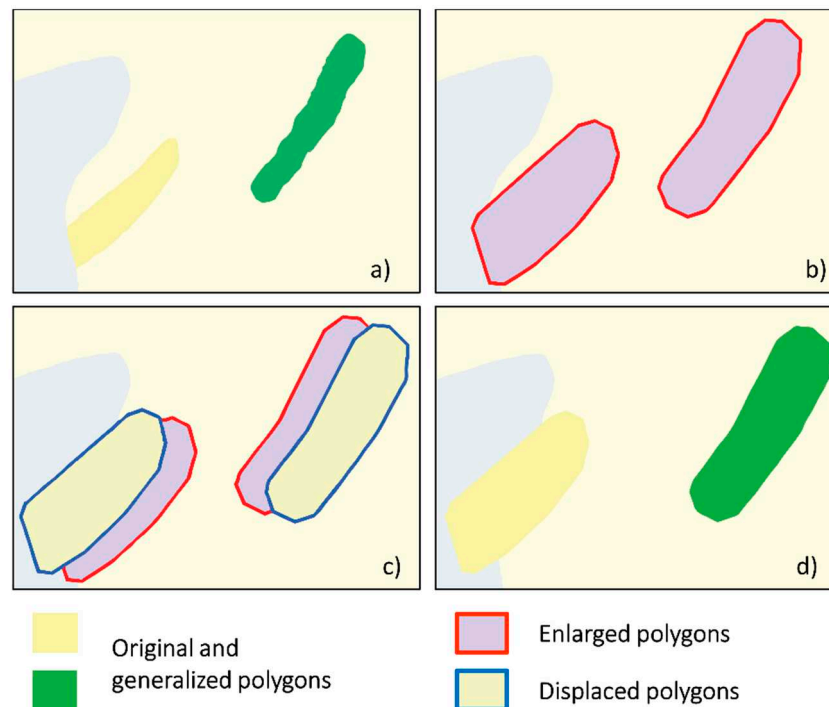


Figure 6. Displacement triggered by polygon enlargement. (a) Original polygons that need to be enlarged. (b) Enlarged polygons, in conflict with object separation. (c) Overlay of enlarged and displaced polygons. (d) Displaced polygons.

We devised two algorithms to implement the displacement operator: a simple pairwise displacement algorithm used when only two polygons are in conflict, and a Voronoi-based algorithm for cases involving multiple polygons. Note, however, that these could be replaced by other, more sophisticated algorithms featuring different displacement characteristics if so desired [39,40].

The pairwise displacement algorithm was inspired by [41] and considers a pair of polygons that are too close to each other to be distinguishable and must, therefore, be pushed back to the object separation distance. The algorithm finds the nearest points between the two polygons, calculates the movement vectors for each polygon, and displaces the polygons from each other (Figure 7). The size of the polygons controls the degree of displacement of each polygon: the smaller the polygons, the more they move, while larger polygons move less. In the example in Figure 7, the polygon with an area of 8666 m² moves only 5 meters, while the polygon with an area of 2471 m² is displaced 18 meters from its initial position.

If multiple polygons are in conflict, the pairwise displacement algorithm cannot guarantee that the minimum separability constraint is met everywhere, as it is unable to safeguard against possible knock-on effects when, e.g., a polygon is pushed into another, neighboring polygon that would then also need to be displaced. Therefore, conflict zones involving three or more polygons are handled

and are achieved by a Voronoi-based displacement approach inspired by algorithms using Voronoi or Voronoi-like cells to control the displacement [42,43].

The Voronoi-based method is divided into three steps. First, polygons are identified that violate the object separation limit (Figure 8a). In the second step, polygons are collapsed to their centroid points, which serve as seed points for the Voronoi diagram (Figure 8b). From the perpendicular bisectors of the Voronoi polygons (i.e., the edges of the Delaunay triangulation), the displacement vectors are identified for each polygon pair. Finally, the polygons are pushed to the minimum separation distance while forcing the polygons to remain within their Voronoi polygon (Figure 8c).

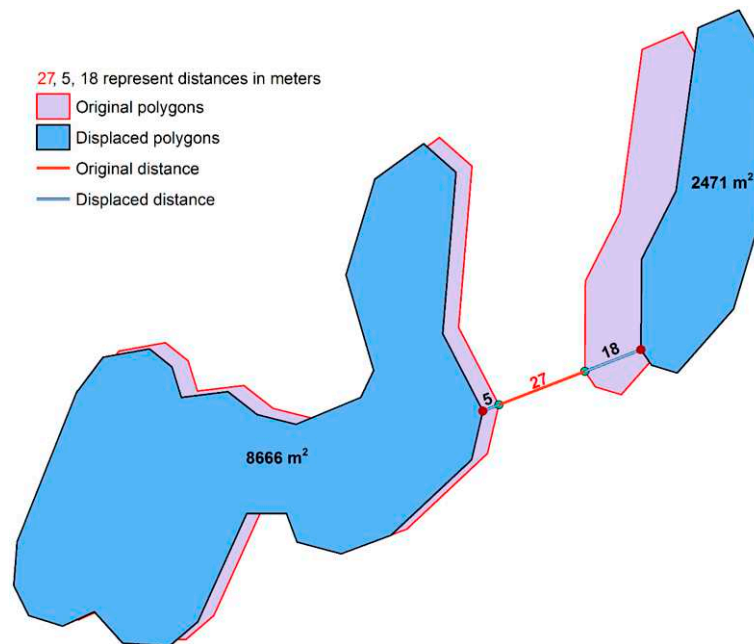


Figure 7. Pairwise displacement: Finding the closest points and calculating the displacement vector based on the proportion the polygons contribute to propagating the displacement.

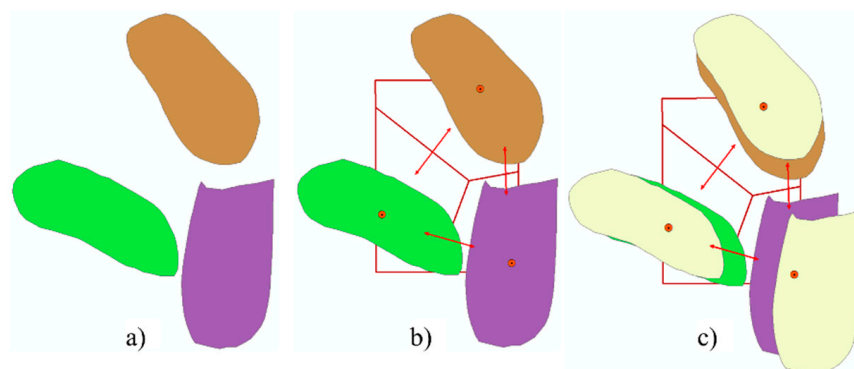


Figure 8. Voronoi-based algorithm. (a) Original polygons. (b) Generation of Voronoi polygons used to propagate displacement. (c) An overlay of the original and displaced polygons.

Figure 9 features two examples of the application of the Voronoi-based displacement algorithm with both the original and displaced polygon positions shown.

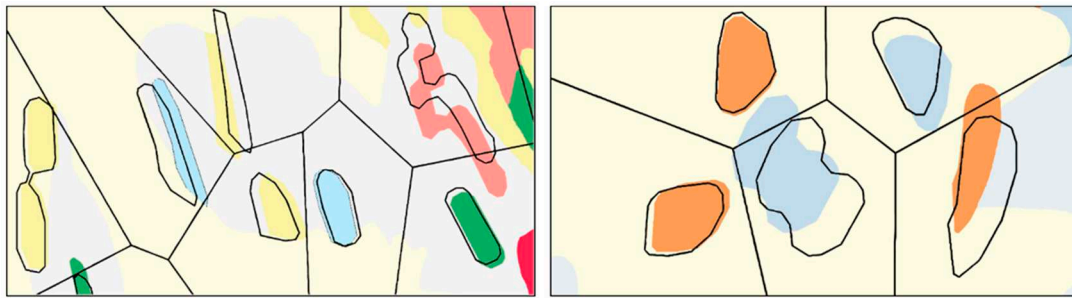


Figure 9. Two examples of the Voronoi-based approach: polygons conflicting with the minimum distance constraint are displaced to the minimum separability distance.

The completion of the displacement operator also marks the end of the process laid out in Figure 4. This process was initially triggered by the minimum area constraint and eventually ensured that this constraint as well as the object separation constraint were fulfilled. All that is left now are the finishing touches by simplification and smoothing of the polygon outlines to ensure the “consecutive vertices” and “outline granularity” constraints are also met (“Output polygons” step in Figure 4). This step, however, uses standard algorithms with conservative parameter settings and thus does not need to be described in detail here.

4.5. Implementation

The proposed workflow uses GeoPandas 0.6.0, an open-source project that handles geospatial data in the Python programming language. Geopandas methods such as dissolving, buffering, centroid identification, convex hull, rotation, scaling, and translation are used to remove, enlarge, aggregate, and displace polygons. Moreover, the ArcPy package from Esri, Inc., is used, implementing ArcGIS toolbox capabilities such as calculating area or finding near and nearby objects. Map visualization is undertaken using Esri ArcGIS 10.6 and Matplotlib, a plotting library for the Python programming language.

5. Experiments and Results

5.1. Data

The data used for our experiments were taken from the “Euriowie Block (including part of Campbells Creek)” 1:25,000 scale geological map published in the year 2000 by the Geological Survey of New South Wales (NSW), Department of Mineral Resources, Australia (<https://www.resourcesandgeoscience.nsw.gov.au>). A portion of this map is shown in Figure 10, left. The “Euriowie Block” is part of the bigger geological block called the “Broken Hill Block”, located near the mining city of Broken Hill, NSW. It contains significant mineral resources, particularly abundant and rich lead–silver–zinc (Pb–Ag–Zn) deposits. The Broken Hill Block’s geology is highly complex, and historically there has always been extensive exploration activity in the area [44,45], which gave us grounds for choosing this particular map. Moreover, the map is openly available for public and research purposes and thus promotes the reproducibility of the results.

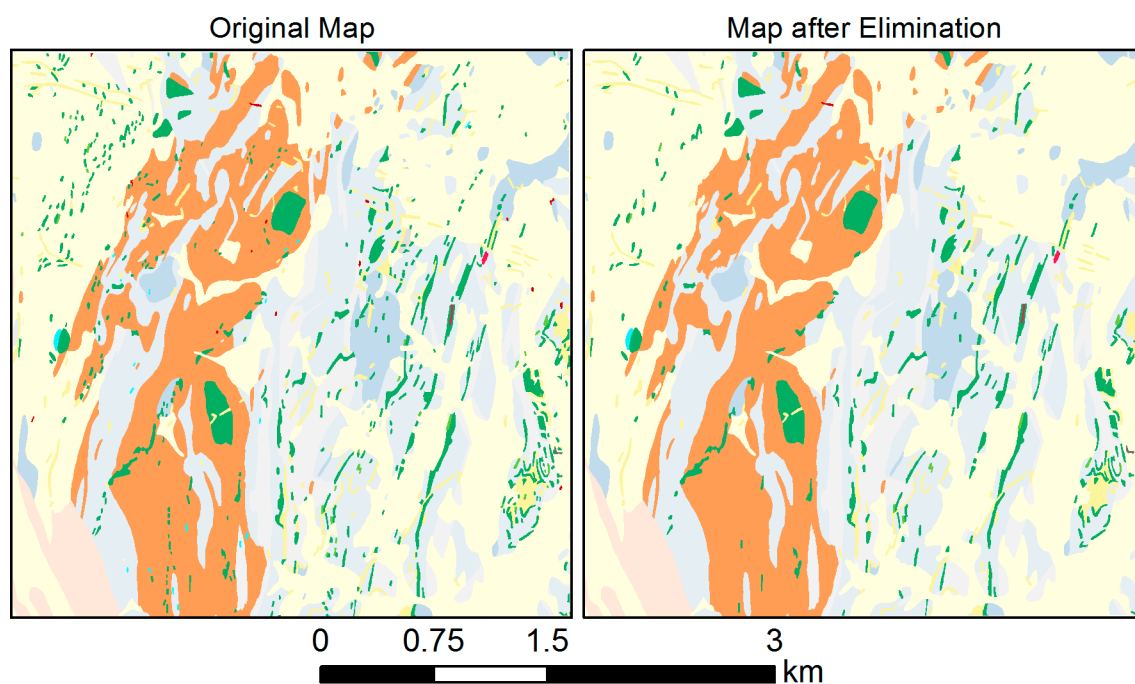


Figure 10. Part of the geological map of the Euriowie Block, NSW, Australia. The original map, left, was produced at the scale of 1:25,000, but is shown here at the scale of 1:50,000, as are all the maps in Figures 10–13. On the right, a map after elimination of less important polygons using the “Radical Law selection” approach is shown.

The map’s objective is to present a broad distribution of rock types, especially those with mineralization or stratigraphic significance. The data are of moderate to high quality, reliability, and internal consistency. It has an accuracy that is appropriate for a map of a scale of 1:25,000.

The geological map consists of 215 sub-categories of rock types or units, which in turn are part of 24 higher categories. Small-sized minerals such as the “hornblende” and “feldspar” groups make up approximately 60% of terrestrial rocks by weight. These mineral groups are mostly related to plutonic (also called intrusive) igneous rocks such as pegmatites (shown in yellow in Figure 10), which form deep under the Earth’s surface and are created by the solidification of magma [46]. They are also present in many types of metamorphic rocks, such as amphibolite (shown in green in Figure 10). Following erosion, they become exposed at the Earth’s surface, often in small patches, represented as small polygons on geological maps. Since the amphibolite and pegmatite units have more mineral deposits and are thus of higher economic value, they will be retained longer in the scale reduction process. Since they often form small polygons, these rock types are good candidates for dealing with minimum size constraints, which are the focus of this paper.

5.2. Elimination

In the sequence of experiments reported in Sections 5.2–5.5, we use goal values for the MA constraint of 0.5×0.5 mm and for the object separation constraint the value of 0.4 mm given in [30]. The whole Euriowie map database consists of $n_s = 1877$ polygons at the source scale of 1:25,000. With a target scale of 1:50,000, the number of polygons n_{MA} that fall below the MA limit is 1135, given a minimum area of 0.5×0.5 mm on the map or 625 m^2 on the ground; (Table 1). This means that almost 60% of all polygons are smaller than what may be defined as the legibility limit. Using the “Radical Law selection” strategy for the elimination operator (Section 4.4.2), the number of polygons that should

be retained, n_t , can be computed as follows (where N_s is the denominator of the source scale and N_t the denominator of the target scale):

$$n_t = n_s \sqrt{N_s/N_t} = 1877 * \sqrt{25,000/50,000} = 1327 \quad (4)$$

That is, according to the Radical Law [33], $n_t = 1327$ polygons should be retained, or conversely, $n_{elim} = n_s - n_t = 550$ polygons should be eliminated. Finally, the number of polygons falling below the MA constraint that should be kept, n_{keep} , can be obtained as $n_{keep} = n_{MA} - n_{elim} = 585$.

Thus, with the Radical Law selection strategy, $n_{elim} = 550$ polygons with an area below the MA threshold should be eliminated from the map, while $n_{keep} = 585$ polygons are also smaller than the MA threshold, but are deemed sufficiently important to be still kept on the map, and thus forwarded to the subsequent generalization operators such as enlargement.

To determine which polygons are eliminated, their importance value is established based on their area as well as their position in the “geological hierarchy” (Table 2). The area values occurring among the n_{MA} polygons smaller than the MA limit range from 7 m² to 625 m². The geological hierarchy follows the order of geological units according to their age: the older the rocks, the more important they are considered, as they likely have more mineral content. Overall, in the Euriowie Block, 15 geological units can be found, which are assigned a value of 15 for the oldest unit and 1 for the youngest unit, respectively (Table 2). For each polygon, its area and its importance value, respectively, are then normalized and summed to obtain the integrated importance value per polygon. After sorting in ascending order, the n_{elim} (550) least important polygons are then terminally eliminated using the `geopandas.dissolve()` algorithm of GeoPandas 0.6.1, which dissolves selected polygons into those neighboring polygons, with whom they share the largest border.

Figure 10, right, shows an excerpt of the Euriowie map after the elimination operation using the Radical Law selection approach compared to the original map, on the left. Examples using the other two selection strategies of the elimination operator will be presented in Section 5.6.1.

Table 2. Assigning importance values to polygons.

#	Property	Value Measurement	Normalized Importance
1	Area of polygons	7–625 m ²	0.0–1.0
2	Geological hierarchy	1–15	0.0–1.0

5.3. Enlargement

Those polygons that are smaller than the MA threshold but sufficiently important to be kept on the target map ($n_{keep} = 585$) need to be enlarged until they reach the minimum area (Section 4.4.3). Depending on the IPQ shape index of polygons, two separate algorithms, buffering and scaling, are used for enlargement.

Overall, 234 polygons (roughly 40%) out of 585 polygons had IPQ values lower than 0.5, indicating rather elongated shapes, and were thus enlarged by buffering. Accordingly, 351 polygons (around 60%) had IPQ values above 0.5, indicating rather round shapes, and were thus enlarged by scaling. The result of the enlargement step is shown in Figure 11.

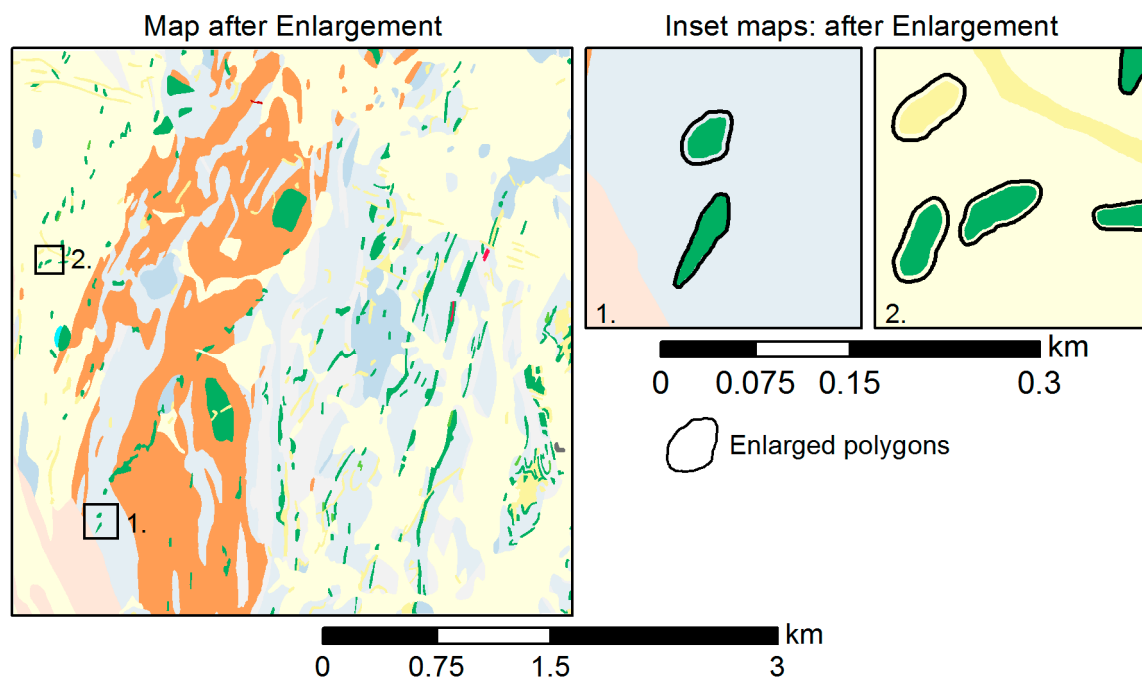


Figure 11. Part of the map after the enlargement of the number of polygons falling below the minimum area (MA) constraint that should be kept (n_{keep}), including two detailed examples illustrating enlarged polygons.

5.4. Aggregation

As the next operation, aggregation combines polygons of the same category (i.e., same geological unit) that fall within the object separation (OS) distance by applying the concave hull algorithm (Section 4.4.4). We used $OS = 0.4$ mm (Table 1) corresponding to 20 meters on the ground at scale 1:50,000 and $k = 3$ in the concave hull algorithm to produce the result depicted in Figure 12. Overall, of the $n_{keep} = 585$ polygons, 78 were aggregated, leaving 507.

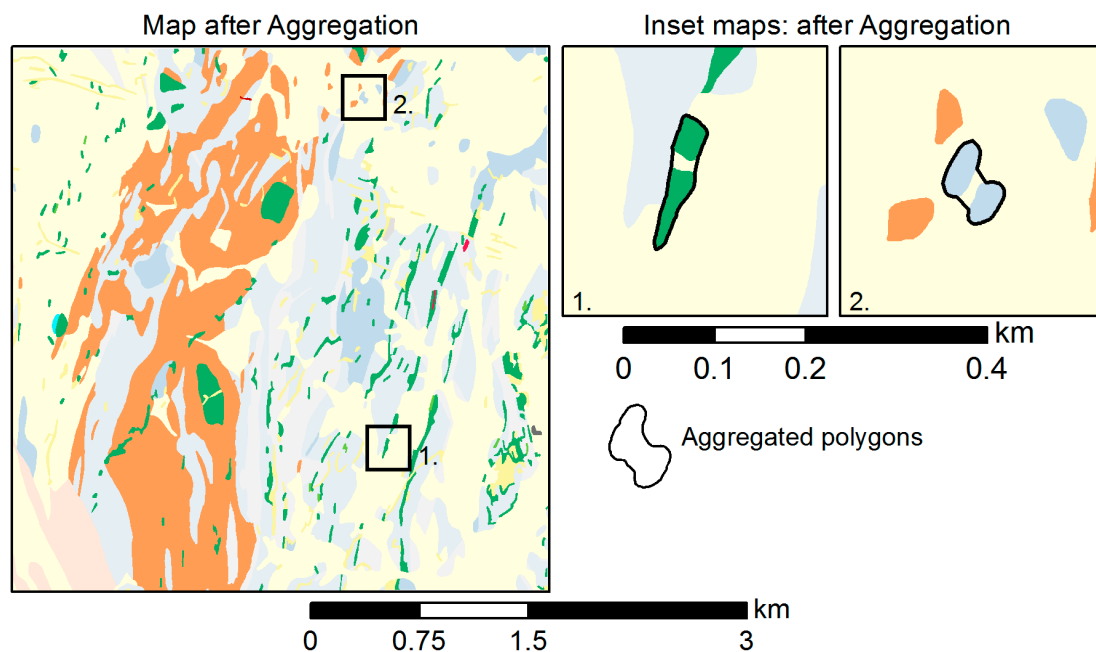


Figure 12. Part of the map after aggregation of polygons closer than the object separation (OS) threshold, including two detailed examples illustrating aggregated polygons.

5.5. Displacement

As a final step, the displacement operator is executed to relocate polygons that are too close to each other (i.e., not meeting the OS limit) from their original position to a minimum separability distance. An excerpt from the result of this operation, and thus the final generalization product, is shown in Figure 13 in comparison to the source map. Overall, 35 polygons were displaced, 24 using the pairwise displacement algorithm and 11 using the Voronoi-based approach.

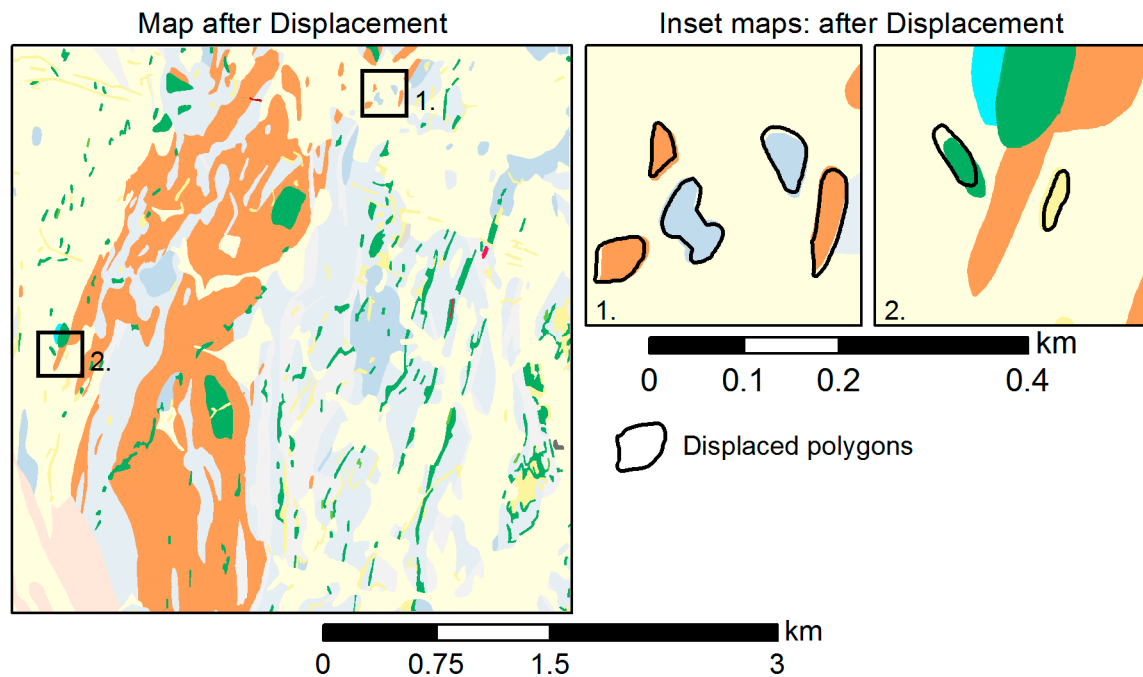


Figure 13. The generalized map after displacement. To the right, two detailed examples illustrating displaced polygons are shown.

5.6. Sensitivity to Parameter Settings

The operators and algorithms of the proposed methodology are controlled by a variety of parameters, all of which can take different values depending on the purpose and application domain of the map in question. Even for the various size constraints, although we defined goal values in Table 1, no canonical or “right” values exist. Hence, in three test cases, we illustrate the sensitivity of the generalization results in variations of the control parameters. Note that in all test cases, no simplification and smoothing was applied. These operators would normally be applied using conservative parameter settings and thus have little effect on the final result (Section 4.4.1).

5.6.1. Test Case 1

In the first test case, the influence of using the three different selection methods for the elimination operator was tested on 1877 polygons contained in the source map at 1:25,000 scale as shown in Figure 14, in the transition to a target scale of 1:50,000. This experiment used the same goal values for the MA and OS constraint as in Sections 5.2–5.5. In the “Radical Law selection”, 29.29% or 563 of the source polygons were removed (Table 3). In “area loss–gain selection”, only 139 were eliminated. Finally, “category-wise selection” resulted in the removal of 417 polygons. Figure 14 allows the visual comparison of the selection methods with each other and the original map. The figure also includes four inset maps for a close-up comparison of the results.

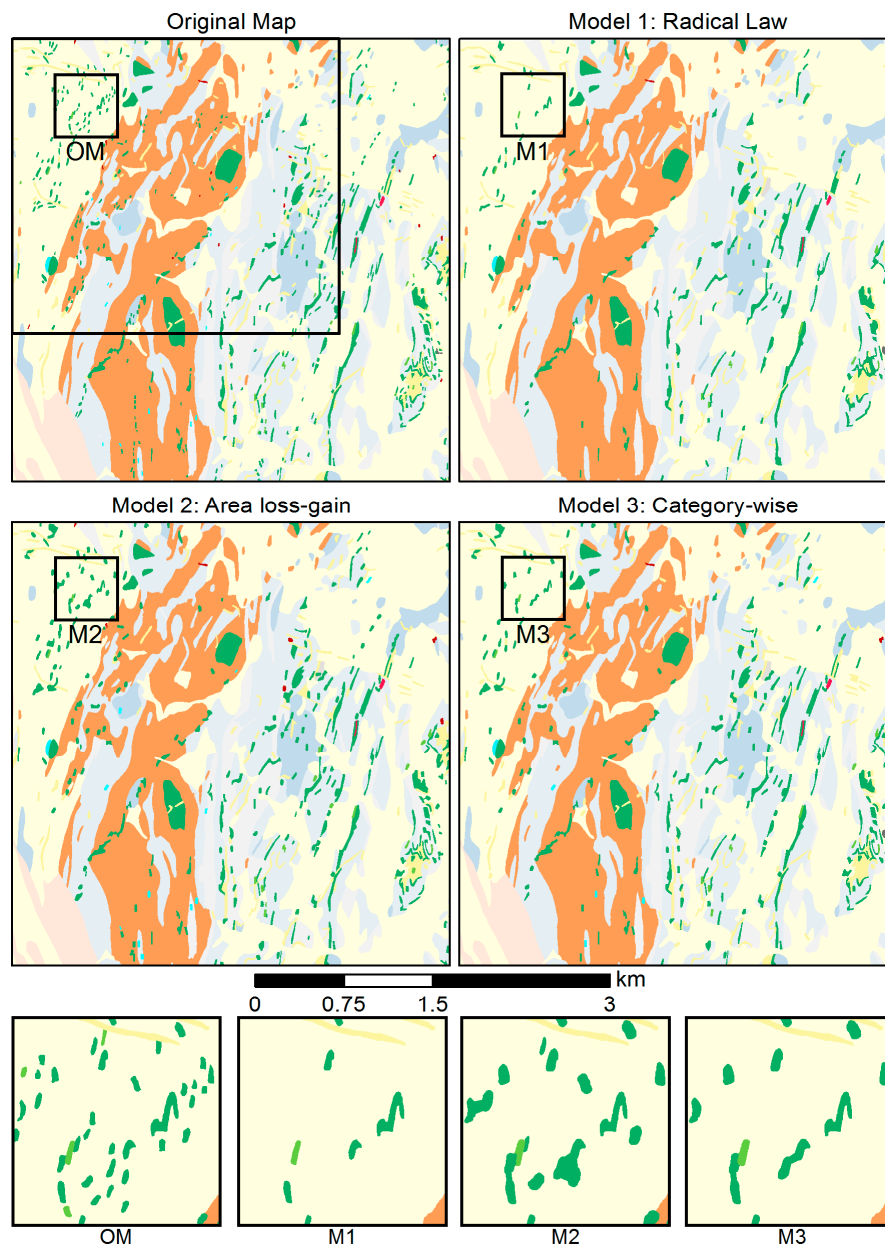


Figure 14. Effect of the selection method in the elimination operator. The large black rectangle in the original map denotes the map window used in Figure 16. The display scale is 1:50,000. The bottom row shows four magnified examples: OM (original map), M1 (Model 1), M2 (Model 2), M3 (Model 3).

Table 3. Comparison of methods of selection in the elimination operator.

Selection Method	Number of Polygons		
	Source Map (25,000)	Target Map (50,000)	Removed
Radical Law selection	1877	1314	563
Area loss–gain selection	1877	1738	139
Category-wise selection	1877	1460	417

5.6.2. Test Case 2

In the second test, we compared three different sets of thresholds for the size constraints, called “Fine-grained” (FG), “Compromise” (CM), and “Coarse-grained” (CG) (Table 4). The first threshold set was inspired by [30]. While their fine-grained goal values were defined for regularly shaped polygons

on topographic maps, the authors noted that larger values should be used for tinted and irregularly shaped polygons, such as those typical of categorical maps. The Coarse-grained constraint set was positioned at the other end of the granularity scale and was obtained from [31], a source devoted to the symbolization of geological maps, defining the minimum size of tinted polygons to be around 2 mm². The Compromise constraint set tries to balance between the Fine-grained and the Coarse-grained sets. The visual comparison of the generalization results produced using the three constraint sets is given in Figure 15. The figure also presents a close-up comparison of the three approaches and the original map.

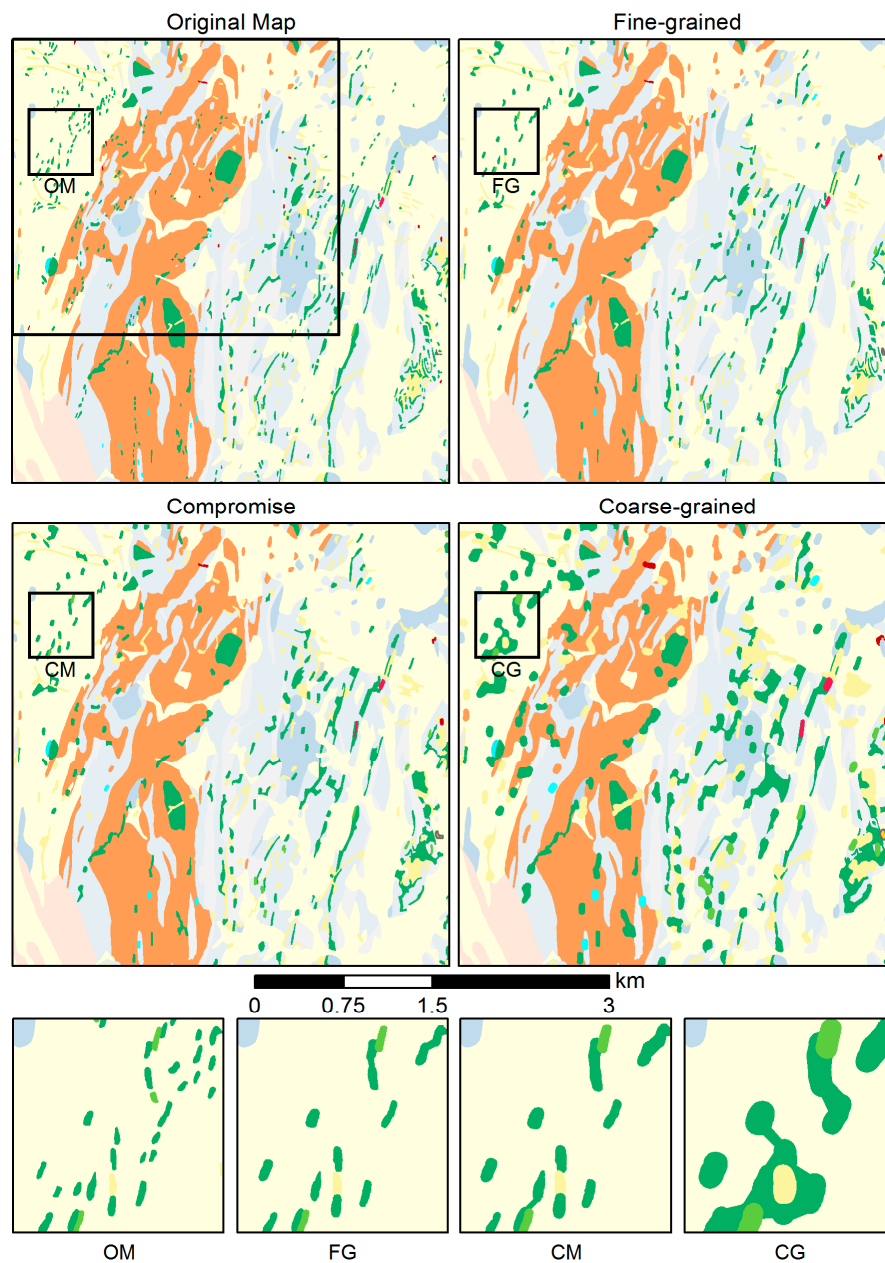


Figure 15. Effect of varying the thresholds for the size constraints using the three sets of goal values from Table 4. The large black rectangle in the original map denotes the map window used in Figure 16. The display scale is 1:50,000. The bottom row depicts four magnified examples: OM (original map), FG (fine-grained), CM (Compromise), CG (coarse-grained).

Table 4. Goal value sets for size constraints used in Test Case 2. MA: minimum area; OS: object separation, #poly = number of polygons.

Threshold Set	MA	Avg. Area	#Poly	OS	Source
Fine-grained (FG)	0.5 × 0.5 mm	19,792 m ²	1265	0.4 mm	[30]
Compromise (CM)	0.75 × 0.75 mm	20,572 m ²	1217	0.6 mm	compromise
Coarse-grained (CG)	2 × 2 mm	26,244 m ²	954	1.0 mm	[31]

5.6.3. Test Case 3

As a final test, we used the Compromise goal values set and the Category-wise selection method, which had both been found to perform best in the above tests, to produce a series of maps at the scales of 1:50,000, 1:100,000, and 1:200,000, generalized from the original 1:25,000 scale map. The interest of this test was in exploring the scale range over which the proposed methodology could be usefully applied. The visual comparison of the resulting generalization series with the original map is presented in Figure 16. Table 5 shows the evolution of the number of polygons and their combined area of four selected geological units present in Figure 16: amphibolite, pegmatite, late Proterozoic, and the soil–sand–gravel–clay unit. Amphibolite (green polygons) and pegmatite (yellow polygons) were chosen because they represent two classes that occur both mainly as small, frequent polygons, and are of high importance due to their age. Late Proterozoic (orange-brown) and soil–sand–gravel–clay (light yellow) units, on the other hand, are young and of the youngest age, respectively, and populate the study area as few but large polygons.

Table 5. Effect of scale reduction using the Compromise set of goal values for the MA and OS constraints (Table 4). The numbers are given for the map window of Figure 16.

Scales	Geological Units	Total Area in m ² (%)	Polygons
Original (1:25,000)	All units	25,032,549 (100%)	1877
	Amphibolite	1,177,891 (4.71%)	838
	Pegmatite	915,132 (3.66%)	393
	Late Proterozoic	713,420 (2.85%)	10
	Soil–sand–gravel–clay	12,371,617 (49.42%)	18
1:50,000 MA—1406.25 m² OS—30 m	All units	25,032,549 (100%)	1349
	Amphibolite	1,382,525 (5.52%)	587
	Pegmatite	888,572 (3.55%)	276
	Late Proterozoic	713,020 (2.85%)	9
	Soil–sand–gravel–clay	12,333,353 (49.24%)	17
1:100,000 MA—5625 m² OS—60 m	All units	25,032,549 (100%)	953
	Amphibolite	1,584,974 (6.32%)	419
	Pegmatite	1,317,895 (5.26%)	197
	Late Proterozoic	697,109 (2.78%)	5
	Soil–sand–gravel–clay	11,857,308 (47.31%)	14
1:200,000 MA—22,500 m² OS—120 m	All units	25,032,549 (100%)	667
	Amphibolite	1,766,809 (7.05%)	294
	Pegmatite	3,126,179 (12.48%)	138
	Late Proterozoic	673,570 (2.69%)	5
	Soil–sand–gravel–clay	10,120,376 (40.40%)	7

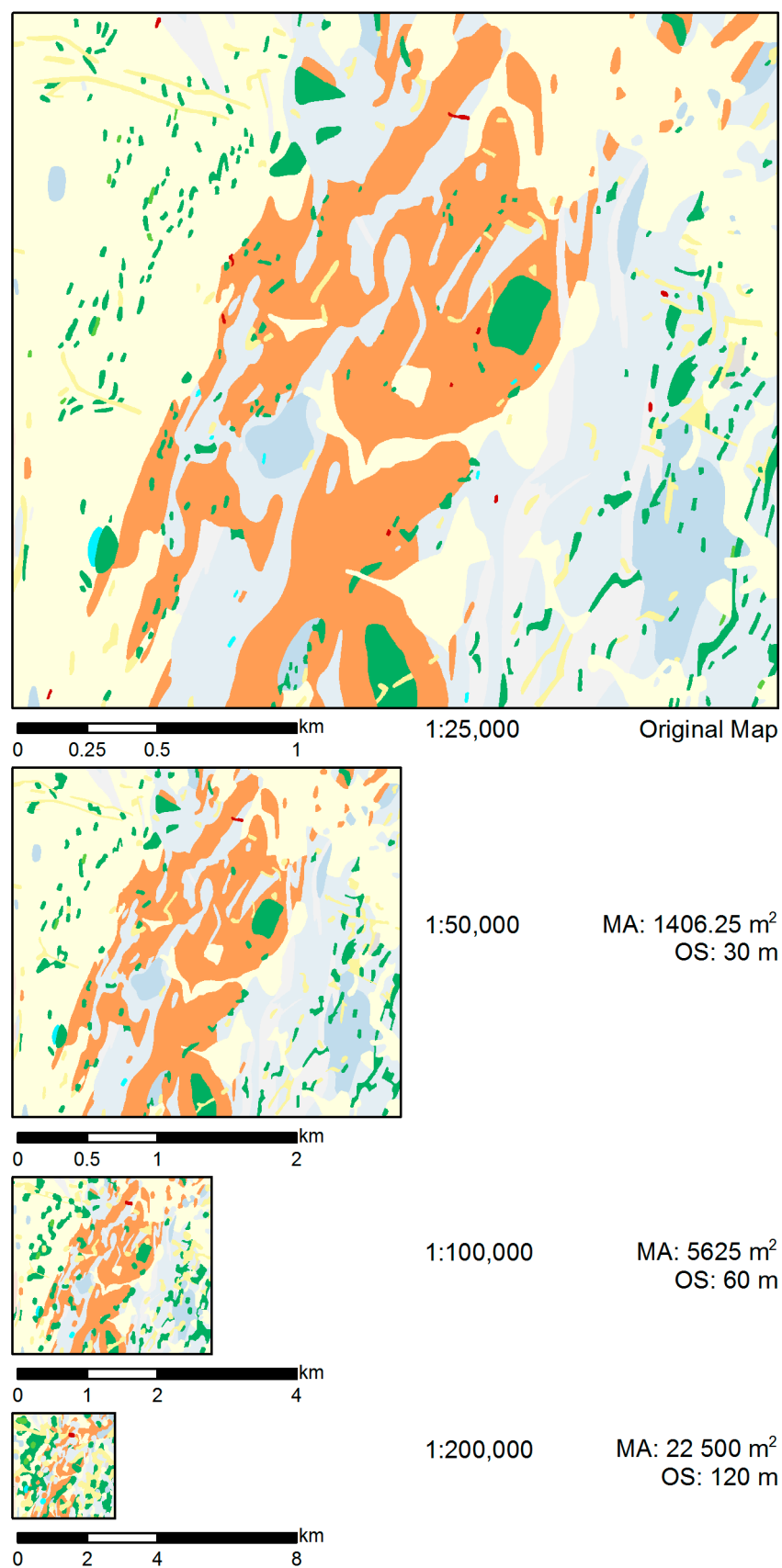


Figure 16. Scale series produced using the Compromised set of goal values for the MA and OS constraints.

5.6.4. Comparison with Cellular Automata Approach

In this section, we compare the results that can be obtained by the proposed methodology with those of an approach based on cellular automata (CA) [23,24], as implemented in the GeoScaler toolbox [47]. As mentioned in Section 2, this approach can probably be considered as the state of the art of geological map generalization available in practice, which is why we used it as a benchmark for comparison. The result of our constraint-based approach was generated using the CM goal values and the category-wise selection method (Figure 17, left). This result is equivalent to the map shown in Figure 15 (bottom left). Since GeoScaler is based on a completely different approach (raster-based, using a CA) as compared to our methodology (which works on individual polygons), it is hard to exactly match the parameter settings for the two methods. GeoScaler also offers a variety of postprocessing tools to further enhance the initial result of the CA algorithm. In order to be able to compare the results of the two methods on a level playing field, we decided not to make use of the GeoScaler postprocessing options and limit the parameter settings to those that find a correspondence between the two methods (i.e., parameters that relate to size constraints). GeoScaler [47] first asks the user for the source and target scale (1:25,000 and 1:50,000 in this case), which then automatically sets the scale reduction factor and size-related parameters. Furthermore, we rasterized the polygon map using a cell-size of 1 meter, the highest possible, to keep the map quality as high as possible. The CA was then run over the rasterized map with a Moore's neighborhood of radius 3 and 2 iterations of final CA generation. Following the CA processing, the raster map was converted back to a vector map, yielding the result shown in Figure 17 (right).

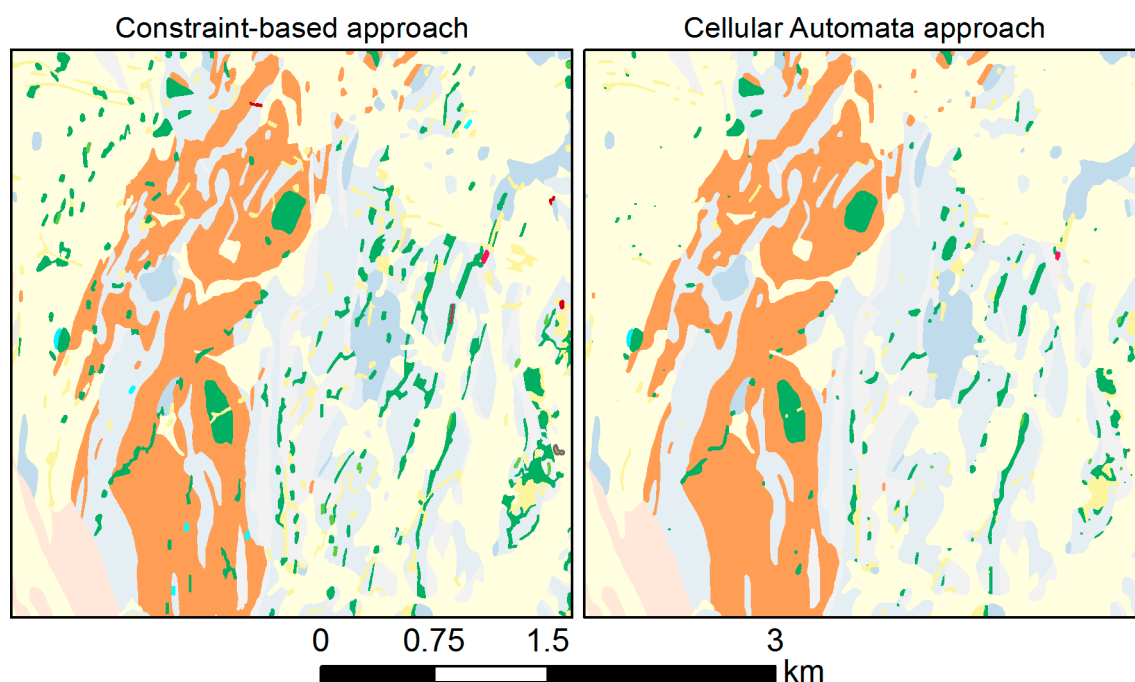


Figure 17. Comparison of two approaches: Constraint-based approach (left) and the cellular automata solution generated in GeoScaler (right).

6. Discussion

This paper presented a methodology for geological map generalization using a constraint-based approach. The main objective of this paper was to demonstrate the usefulness of the constraint-based approach in map generalization when applied to geological maps, not to solve geological map generalization comprehensively (more feature classes are addressed in [23,24]). This was done against the background that the constraint-based approach has so far not been explicitly applied to the automated generalization of geological maps. Hence, the proposed methodology focused on a

sub-problem of geological map generalization, though nevertheless an important one. In particular, it focused on size constraints as the primary driving force of maintaining map legibility through map generalization. Furthermore, we focused on the case of small, free-standing polygons (or area patches; [28]), as they represent a frequent case not only in geological maps but also other types of categorical maps such as soil maps or land cover maps.

Based on the experiments reported in Section 5, we can make several general observations:

1. The proposed methodology resolves the main legibility problems associated with small polygons in a step-by-step manner. The resulting map is more readable, and map features remain distinguishable after generalization (Figure 13).
2. Although the goal values of the size constraints are defined globally, each polygon is treated individually to its own, specific properties rather than by a global process such as cellular automata. Hence, by consideration of the semantics of individual polygons, important polygons can be protected by enlargement; by consideration of shape properties, the shape characteristics of the individual polygons are largely maintained.
3. The approach, by being based on testing whether the goal values of constraints are met, allows self-evaluation, and can guarantee that the legibility limits are met—but not more than that. Again, this differs from global approaches such as cellular automata [23,24].
4. Few parameters are required to control the generalization methodology. The process is initially triggered by the MA constraint and further assisted by few additional size constraints (most importantly, the OS constraint). Once the goal values of the size constraints and additional algorithm-specific parameters have been set, the methodology operates automatically, without further human intervention.
5. The constraints' goal values are, first of all, a function of the map legibility at the target scale, and hence allow adapting to the desired scale transition. Furthermore, the goal values also allow for controlling the overall granularity of the output map (Table 4 and Figure 15), depending on the map purpose.
6. Despite the rather low number of constraints and control parameters, the methodology is modular and features several generalization operators, thus achieving considerable flexibility.

Because the proposed methodology builds on the universally valid concept of constraints relating to map legibility, the potential exists to apply the same approach to the generalization of other categorical maps, such as vegetation or soil maps. Furthermore, since the workflow is modular, any of the algorithms used in our methodology could be replaced by another algorithm that is perhaps better suited for the peculiarities of the given generalization problem. Thus, for instance, the aggregation or displacement algorithms used in this paper could be replaced by more elaborate algorithms if so desired.

Using a workflow approach, we showed how the proposed methodology could be implemented using the libraries of a general-purpose GIS, achieving a behavior similar that of an agent-based system, but with less implementation effort and less effort for parameter tuning.

Comparing Figures 10–13, which present the results of the individual generalization operators, it becomes obvious that elimination and aggregation have the most far-reaching effects. Elimination may retain, or alternatively destroy, both the patterns of spatial arrangements of polygons, as well as the balance of area proportions of the different categories appearing on the map. Below, we discuss the available elimination algorithms separately through Test Case 1. Aggregation is the other operator that can lead to significant alteration of the map image, in this case however by area gain and shape modification of the polygons concerned. As Figure 12 suggests, the aggregation algorithm that was implemented in our workflow can sometimes lead to undesirable shape and area changes, and would be better replaced by a typification algorithm [36,37], which however would require preceding detection of group patterns amenable to typification [10].

Elimination is the first but also the most radical step of the methodology, as some polygons are permanently removed from the target map. Selective elimination of small island polygons supported by importance values gives way to preserve polygons that are important relative to others, thus reflecting geological importance or economic value. However, in the elimination process, the connectivity between neighboring polygons is not considered, which may lead to the removal or uncontrolled dissolution of certain group patterns (e.g., clusters, alignments) of polygons. Examples of this effect can be observed comparing Figure 10, left (the original map), and Figure 10, right (the map after elimination).

To address this issue, we ran Test Case 1 on three selection methods (Section 5.6.1): Radical Law selection, area loss–gain selection, and category-wise selection. As Figure 14 and Table 3 show for the scale transition to 1:50,000, area loss–gain selection preserved the number and structure of polygons best. In contrast, Radical Law selection had the most destructive effect on polygon groups. However, when the scale transition is larger, e.g., with a target scale of 1:100,000 or 200,000, more polygons will have to be removed to balance the area gain induced by enlargement. Thus, category-wise selection seems to be the better approach for the elimination operation in greater scale reductions, as it distributes the removal of polygons evenly across each category. In general, Radical Law selection is not recommended for use in categorical map generalization unless more strongly generalized (“overgeneralized”) maps are desired. One of the essential requirements of categorical map generalization is to preserve areas as much as possible across scale ranges, which can be achieved using the area loss–gain balance selection method. Based on Test Case 1, however, the category-wise selection method ultimately seems more appropriate as it evens out the elimination of polygons across categories, and also represents a good compromise. Nevertheless, regardless of which of the three selection methods is used, none can safeguard against the inadvertent destruction of group patterns, especially not at greater scale reductions. The actual generalization stage should thus be preceded by a stage devoted to the recognition of essential collective patterns present in the source map [16,18], which led us to develop a process for the recognition of polygon group patterns [10].

In Test Case 2, documented in Figure 15 and Table 4, we compared three different sets of goal values to explore their effect on the granularity of the resulting map. The most fine-grained of the three sets (FG) was originally developed for topographic mapping [30]. As Figure 15 shows, it is however reaching its limits for the generalization of geological maps that are usually populated by tinted, irregularly shaped polygons that often have reduced visual contrast (as is for instance particularly the case with the yellow pegmatite polygons). For such maps, the goal values of the FG set seem too detailed. Thus, in [31] the use of considerably higher goal values for geological maps was proposed, which produces a rather coarse-grained result with more polygons being enlarged and aggregated. As Figure 15 suggests, this Coarse-grained (CG) set of goal values may lead to overgeneralization and excessive loss of detail, particularly when the focus is on small polygons, as is the case in this paper. This is confirmed by the average polygon area and the number of polygons (Table 4), which are markedly different from those of the other two sets of goal values. The Compromise (CM) set of goal values allows for a more differentiated picture to be maintained, while ensuring that legibility is maintained. Note that in this set, the values were chosen to lie closer to those of the FG set than those of the CG set, with the intention of retaining as much detail as possible, thus also allowing a greater range of scale reduction. This is also reflected by the results reported in Table 4 for the average polygon area and the number of polygons, which are very similar to those obtained with the FG set.

In Test Case 3, we used the Compromise goal values and the Category-wise selection strategy to produce a series of generalized maps, shown in Figure 16 at the corresponding target scales. While the resulting map at 1:50,000 overall looks convincing, the map at 1:100,000 starts showing signs of visual imbalance. At the scale of 1:200,000, the map breaks down completely. The visual impression is supported by Table 5, which shows that the relative area per class becomes heavily imbalanced after 1:100,000. In the original map, the Amphibolite and Pegmatite categories are the most frequent ones regarding the number of polygons, which is due to the fact that they mostly occur as small polygons.

Hence, they are candidates for removal due to their generally small size, and their number decreases significantly in the transition from 1:100,000 and 1:200,000 (Table 5). However, as they are so numerous, these two categories are also most affected by enlargement and particularly by aggregation; hence, the area gained in the generalization process is disproportional. Again, this effect suggests that beyond a scale of 1:100,000 the results are no longer pleasing. Once again, this points to the necessity of an approach that is based on the recognition of local group structures which can then inform improved contextual generalization operators [10].

Figure 17 compares the proposed constraint-based methodology and the CA approach of GeoScaler [23,24], and both methods indeed yielded comparable results in that the legibility was improved. However, the CA approach, as it is based on a moving-window operator that uses the same, essentially majority-based principle across the entire map, has a tendency towards dilating larger polygons while eroding small polygons. Hence, more small polygons disappeared than in the result of the proposed methodology, and some polygons became hardly visible. GeoScaler includes post-processing operations, such as enlargement of too small polygons, which can be applied to ensure the legibility of all polygons. However, even that postprocessing operation will not bring back those small polygons that have been removed. Thus, overall the proposed constraint-based methodology seems to outperform the CA-based approach regarding the adequate maintenance of small polygons. Additionally, due to the raster-based majority filtering of the CA approach, characteristic polygon shapes also become more uniform than in our proposed methodology.

We have already mentioned the lack of explicit polygon group recognition in the elimination operator as the first shortcoming of the proposed methodology. Here, we see the second weakness: because the methodology focuses on small polygons and favors small polygons of important geological units by enlarging them, those small polygons grow disproportionately when the scale reduction factor is large. Once again, this points to the necessity of treating groups of polygons rather than individual polygons. While the methodology of this paper includes contextual generalization operators (aggregation, displacement), “context” is merely understood as the immediate, first-order neighborhood defined by the OS constraint. If polygons are separated by a distance slightly exceeding the OS limit, no link is detected. Likewise, there is no facility for detecting higher-order neighbors and hence forming larger, contiguous groups of polygons. Hence, for a full treatment of the small polygons forming the focus of this paper, group pattern detection procedures are needed, as well as contextual generalization operators that can make use of such group patterns. In related work [10], we therefore propose a methodology that considers proximity as well as geometrical similarities (shape, size, and orientation of map features) and attribute similarities to find, refine, and form groups that can be used to subsequently inform contextual generalization operators, such as aggregation and typification, to overcome persistent limitations in generalization approaches such as those shown in this paper.

7. Conclusions

Taking the example of size constraints (minimum area, object separation, etc.), this paper demonstrated how constraint-based map generalization can be realized for the case of small island polygons, which are mainly dominated by size constraints. Size constraints are rather simple but very fundamental; everything else depends on them. Thus, starting with the most fundamental constraint—the minimum area constraint—we were able to demonstrate how a constraint-based approach can be built, triggering different generalization operators depending on the local analysis of the individual situation found. From a practical perspective, we were able to show how such an approach can be implemented using the libraries of a general-purpose GIS, achieving a behavior similar to that of an agent-based system but with less implementation effort and less effort for parameter tuning.

As our experiments also showed, an approach focusing solely on size constraints and local analysis and operations is not sufficient if greater scale reductions are to be achieved. Thus, future research will have to explore the detection of groups of polygons defined by proximity and similarity (in

attribute, size, shape, orientation, etc.), as well as contextual generalization operators for aggregation and typification that can make use of those group patterns.

Author Contributions: Conceptualization, Azimjon Sayidov and Robert Weibel; Data curation, Azimjon Sayidov; Formal analysis, Azimjon Sayidov; Investigation, Azimjon Sayidov; Methodology, Azimjon Sayidov, Meysam Aliakbarian and Robert Weibel; Project administration, Azimjon Sayidov and Robert Weibel; Software, Azimjon Sayidov and Meysam Aliakbarian; Supervision, Robert Weibel; Validation, Azimjon Sayidov; Visualization, Azimjon Sayidov and Meysam Aliakbarian; Writing—original draft, Azimjon Sayidov; Writing—review and editing, Azimjon Sayidov and Robert Weibel. All authors have read and agreed to the published version of the manuscript.

Funding: The Ph.D. research was supported by a scholarship of the Swiss Federal Commission for Scholarships (FCS) under the Grant 2015.0750.

Acknowledgments: We are grateful for the support provided to the first author by the Swiss Federal Commission for Scholarships (FCS). The authors would like to acknowledge their use of the “Eurioiwie (including part of Campbells Creek)” 1:25,000 geological map provided by the Geological Survey of New South Wales, Department of Mineral Resources, NSW, Australia, which formed the basis for the experiments reported in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sayidov, A.; Weibel, R. Constraint-based approach in geological map generalization. In Proceedings of the 19th ICA Workshop on Generalisation and Multiple Representation, Helsinki, Finland, 14 June 2016.
2. Sayidov, A.; Weibel, R. Automating geological mapping: A constraint-based approach. In Proceedings of the CEUR (Central Europe) Workshop Proceedings, Leeds, UK, 30 October–2 November 2017.
3. Ruas, A.; Duchêne, C. A prototype generalisation system based on the multi-agent system paradigm. In *Generalisation of Geographic Information: Cartographic Modelling and Applications*; Elsevier: Amsterdam, The Netherlands, 2007; pp. 269–283.
4. Weibel, R.; Dutton, G.H. Constraint-based automated map generalization. In *8th International Symposium on Spatial Data Handling*; Taylor & Francis: Columbia, UK, 1998; pp. 214–224.
5. Barrault, M.; Regnaud, N.; Duchene, C.; Haire, K.; Baeijs, C.; Demazeau, Y.; Hardy, P.; Mackaness, W.; Ruas, A.; Weibel, R. Integrating multi-agent, object-oriented and algorithmic techniques for improved automated map generalization. In Proceedings of the 20th International Cartographic Conference, Beijing, China, 6–10 August 2001; Volume 3, pp. 2110–2116.
6. Harrie, L.; Weibel, R. Modelling the overall process of generalisation. In *Generalisation of Geographic Information: Cartographic Modelling and Applications*; Elsevier: Oxford, UK, 2007; pp. 67–87.
7. Stoter, J.; Burghardt, D.; Duchêne, C.; Baella, B.; Bakker, N.; Blok, C.; Pla, M.; Regnaud, N.; Touya, G.; Schmid, S. Methodology for evaluating automated map generalization in commercial software. *Comput. Environ. Urban Syst.* **2009**, *33*, 311–324. [CrossRef]
8. Galanda, M. Automated Polygon Generalization in a Multi-Agent System. Ph.D. Thesis, University of Zurich, Zurich, Switzerland, 2003.
9. Ruas, A. *Modèle de Généralisation de Données Géographiques à Base de Contraintes et D'autonomie*; Université de Marne-la-Vallée: Champs-sur-Marne, France, 1999.
10. Sayidov, A.; Weibel, R.; Leyk, S. Recognition of group patterns in geological maps by building similarity networks. *Geocarto Int.* **2020**, 1–20. [CrossRef]
11. Juliev, M.; Mergili, M.; Mondal, I.; Nurtaev, B.; Pulatov, A.; Hübl, J. Comparative analysis of statistical methods for landslide susceptibility mapping in the Bostanlik District, Uzbekistan. *Sci. Total Environ.* **2019**, *653*, 801–814. [CrossRef] [PubMed]
12. Maine.gov. Maine Geological Survey: Reading Detailed Bedrock Geology Maps. Available online: <https://www.maine.gov/dacf/mgs/pubs/mapuse/bedrock/bed-read.htm> (accessed on 18 July 2019).
13. Jirsa, M.A.; Boerboom, T.J. Components of Geologic Maps. Available online: <http://docslide.us/documents/components-of-geologic-maps-by-mark-a-jirsa-and-terrence-j-boerboom-2003.html> (accessed on 14 January 2020).
14. Dawes, R.L.; Dawes, C.D. *Basics-Geologic Structures*; Geology of the Pacific Northwest: Wenatchee, WA, USA, 2013.
15. Graymer, R.W.; Moring, B.C.; Saucedo, G.J.; Wentworth, C.M.; Brabb, E.E.; Knudsen, K.L. *Geologic Map of the San Francisco Bay Region*; US Department of the Interior, US Geological Survey: Reston, VA, USA, 2006.

16. Steiniger, S.; Weibel, R. Relations among map objects in cartographic generalization. *Cartogr. Geogr. Inf. Sci.* **2007**, *34*, 175–197. [[CrossRef](#)]
17. Downs, T.C.; Mackaness, W. An integrated approach to the generalization of geological maps. *Cart. J.* **2002**, *39*, 137–152. [[CrossRef](#)]
18. Brassel, K.E.; Weibel, R. A review and conceptual framework of automated map generalization. *Int. J. Geogr. Inf. Syst.* **1988**, *2*, 229–244. [[CrossRef](#)]
19. Mackaness, W.A.; Edwards, G. The importance of modelling pattern and structure in automated map generalization. In Proceedings of the Joint ISPRS/ICA Workshop on Multi-scale Representations of Spatial Data, Ottawa, ON, Canada, 7–8 July 2002.
20. Steiniger, S.; Weibel, R. A conceptual framework for automated generalization and its application to geologic and soil maps. In Proceedings of the 22nd International Cartographic Conference, La Coruña, Spain, 11–16 July 2005.
21. Shea, K.S.; McMaster, R.B. *Cartographic Generalization in a Digital Environment: When and How to Generalize*; AUTO-CARTO 9: Baltimore, MD, USA, 1989.
22. McCabe, C.A. Vector Approaches to Generalizing Faults and Polygons in Santa Rosa, California, Case Study. 2007. Available online: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.643.17&rep=rep1&type=pdf>. (accessed on 14 January 2020).
23. Smirnoff, A.; Paradis, S.J.; Boivin, R. Generalizing surficial geological maps for scale change: ArcGIS tools vs. cellular automata model. *Comput. Geosci.* **2008**, *34*, 1550–1568. [[CrossRef](#)]
24. Smirnoff, A.; Huot-Vézina, G.; Paradis, S.J.; Boivin, R. Generalizing geological maps with the GeoScaler software: The case study approach. *Comput. Geosci.* **2012**, *40*, 66–86. [[CrossRef](#)]
25. Bader, M.; Weibel, R. Detecting and resolving size and proximity conflicts in the generalization of polygonal maps. In Proceedings of the 18th International Cartographic Conference, Stockholm, Sweden, 23–27 June 1997; Volume 23, p. 27.
26. Galanda, M.; Weibel, R. An agent-based framework for polygonal subdivision generalization. In *Advances in Spatial Data Handling, 10th International Symposium on Spatial Data Handling*; Richardson, D., van Oosterom, P., Eds.; Springer: Berlin, Germany, 2002; pp. 121–136.
27. Edwardes, A.; Mackaness, W.A. Modelling knowledge for automated generalisation of categorical maps—A constraint based approach. In *Innovations in GIS 7. GIS and Geocomputation*; Atkinson, P., Martin, D., Eds.; Taylor & Francis: London, UK, 2000; pp. 161–174.
28. Müller, J.C.; Wang, Z. Area-patch generalisation: A competitive approach. *Cartogr. J.* **1992**, *29*, 137–144. [[CrossRef](#)]
29. Ruas, A.; Plazanet, C. Strategies for automated generalization. In *Advances in GIS Research II, Proceedings of the 7th International Symposium on Spatial Data Handling*; Kraak, M.J., Molenaar, M., Eds.; Taylor & Francis: London, UK, 1996; pp. 319–336.
30. Spiess, E.; Baumgartner, U.; Arn, S.; Vez, C. Topographic maps map graphics and generalisation. *Cartogr. Publ. Ser.* **2005**, *17*, 121.
31. Subcommittee, G.D. U.S. Geological Survey FGDC Digital Cartographic Standard for Geologic Map Symbolization (FGDC Document Number FGDC-STD-013-2006); U.S. Geological Survey for the Federal Geographic Data Committee: Reston, VA, USA, 2006.
32. Lichtner, W. Computer-assisted processes of cartographic generalization in topographic maps. *Geo Process.* **1979**, *1*, 183–199.
33. Töpfer, F.; Pillewizer, W. The principles of selection. *Cartogr. J.* **1966**, *3*, 10–16. [[CrossRef](#)]
34. Osserman, B.Y.R. The isoperimetric inequality. *Bull. Amer. Math. Soc.* **1978**, *6*, 1182–1238. [[CrossRef](#)]
35. Touya, G. A road network selection process based on data enrichment and structure detection. *Trans. GIS* **2010**, *14*, 595–614. [[CrossRef](#)]
36. Regnauld, N. Contextual building typification in automated map generalization. *Algorithmica* **2001**, *30*, 312–333. [[CrossRef](#)]
37. Burghardt, D.; Cecconi, A. Mesh simplification for building typification. *Int. J. Geogr. Inf. Sci.* **2007**, *21*, 283–298. [[CrossRef](#)]
38. Moreira, A.; Santos, M.Y. Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. In Proceedings of the Second International Conference on Computer Graphics Theory and Applications, Algarve, Portugal, 5–7 March 2011; pp. 61–68.

39. Galanda, M.; Weibel, R. Using an energy minimization technique for polygon generalization. *Cartogr. Geogr. Inf. Sci.* **2003**, *30*, 263–279. [[CrossRef](#)]
40. Bader, M.; Barrault, M.; Weibel, R. Building displacement over a ductile truss. *Int. J. Geogr. Inf. Sci.* **2005**, *19*, 915–936. [[CrossRef](#)]
41. Mackaness, W.; Purves, R. Automated displacement for large numbers of discrete map objects. *Algorithmica* **2001**, *30*, 302–311. [[CrossRef](#)]
42. Ai, T.; Zhang, X.; Zhou, Q.; Yang, M. A vector field model to handle the displacement of multiple conflicts in building generalization. *Int. J. Geogr. Inf. Sci.* **2015**, *29*, 1310–1331. [[CrossRef](#)]
43. Basaraner, M. A zone-based iterative building displacement method through the collective use of Voronoi Tessellation, spatial analysis and multicriteria decision making. *Bol. Ciências Geodésicas* **2011**, *17*, 161–187. [[CrossRef](#)]
44. Barnes, R.G. *Metallogenic Studies of the Broken Hill and Eurioiwie Blocks. 1. Styles of Mineralisation in the Broken Hill Block. 2. Mineral Deposits of the Southwestern Broken Hill Block*; Geological Survey of New South Wales: New South Wales, Australia, 1988.
45. Peljo, M. *Broken Hill Exploration Initiative: Abstracts of Papers Presented at the July 2003 Conference in Broken Hill*; Geoscience Australia, Ed.; Geoscience Australia: Canberra, Australia, 2003; ISBN 978-0-642-46771-3.
46. Wilson, J.R. *Minerals and Rocks*; BookBoon: London, UK, 2010.
47. Huot-Vézina, G.; Boivin, R.; Smirnoff, A.; Paradis, S.J. GeoScaler: Generalization tool (with a supplementary user guide in French). *Geol. Surv. Can.* **2012**, *6231*, 82. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).